Hindawi
IET Information Security
Volume 2023, Article ID 7171600, 8 pages
https://doi.org/10.1049/2023/7171600

Research Article

Similarity Property and Slide Attack of Block Cipher FESH

Yafei Zheng [1,2,3] **and Wenling Wu** [1,3]

[1]*Trusted Computing and Information Assurance Laboratory, Institute of Software Chinese Academy of Sciences, Beijing 100190, China*
[2]*State Key Laboratory of Cryptology, Beijing 100878, China*
[3]*University of Chinese Academy of Sciences, Beijing 100049, China*

Correspondence should be addressed to Yafei Zheng; zhengyafei@iscas.ac.cn

Received 28 July 2023; Revised 9 November 2023; Accepted 21 November 2023; Published 14 December 2023

Academic Editor: Taimur Bakhshi

This paper focuses on similarity properties and extension of the classical slide property of block ciphers. Taking FESH, an award-winning block cipher of the National Cryptographic Algorithm Design Competition 2019, as an example, similarity properties of the encryption and key transformation are found, owing to the similar structures that the encryption and key transformation adopted, and the constants generation. Based on the similarity properties, extended slide properties can be constructed for FESH. Slide attacks of FESH are then proposed. The similarity properties and extended slide property are immune to the increasing of iterated rounds, i.e., it cannot be avoided by increasing the round number of FESH. Furthermore, extended slide property helps relaxing the strict requirements of the subkeys in slide attacks. Taking Feistel and SPN structures as examples, frameworks of slide attacks based on the extended slide properties are presented. Slide attack of FESH is exactly a concrete example of SPN structure.

1. Introduction

FESH [1] is a block cipher submitted to the National Cryptographic Algorithm Design Competition held by the Chinese Association for Cryptologic Research (CACR) in 2019. With excellent design features and implementation performance, FESH finally became an award-winning block cipher of the competition. FESH adopts SPN structure, and the round function takes advantage of bit slice technique. It is worth noting that the key schedule of FESH shares the similar structure with the encryption. In the key schedule, lighter 4-bit Sbox is selected for lightweight purpose. The linear layer of the key schedule adopts a simplified version of the encryption linear layer. To avoid symmetric property, constant addition is introduced. The designers of FESH have evaluated its security against differential attack, boomerang attack, linear attack, impossible differential attack, integral attack, and related key attack. The most powerful attack is the 9-round/13-round related-key/related-key boomerang attack [1]. There is still a long way in terms of the iteration rounds of FESH with sufficient security redundancy. FESH has shown security against existing attacks with no potential security vulnerabilities.

Slide attack, proposed by Biryukov and Wagner [2] in 1999, is an attack reverses an early common cognition of designers: for an iterative block cipher (generally referred to Feistel block ciphers), even under the premise that the round function is relatively weak, as long as the times of iteration is large enough, the cipher can achieve strong security. Slide attack can be regarded as a special case of related key attack [3], and each subkey is the same or the subkey sequence has a short period. In most cases, slide attack will not be affected by the number of iterated rounds. Cryptographers have subsequently given a variety of improved slide attacks. In 2000, Biryukov and Wagner [4] improved the classical slide attack by using techniques complementation slide and slide with a twist, and further the techniques are applied to DES variant and GOST. In 2008, Biham et al. [5] combined the cycle structure of encryption and round function, and slide pairs can be found with lower complexity. Meanwhile, slide pairs satisfying the needs of different types of round functions can be found. The idea is applied to

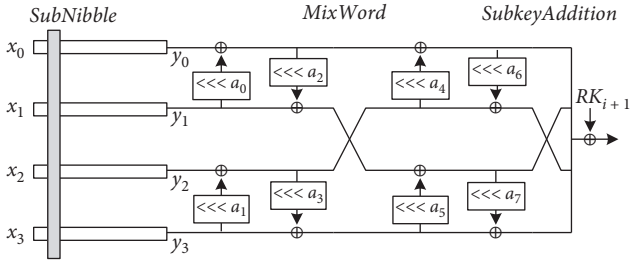

FIGURE 1: Round function of FESH.

TABLE 1: Sbox.

x	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
$S(x)$	3	d	f	a	0	7	c	1	4	2	9	5	b	e	6	8

TABLE 2: Parameters in MixWord.

Cipher	a_0	a_1	a_2	a_3	a_4	a_5	a_6	a_7
FESH-128-128/256	29	13	4	21	15	19	25	6
FESH-256-256	58	33	8	1	17	44	5	9

GOST. In 2013, related key slide attacks on block ciphers with secret components [6] are introduced and applied to lightweight encryption algorithm specified and approved in NRS 009-6-7:2002 [7] by Electricity Suppliers Liaison Committee to be used with tokens in prepayment electricity dispensing systems in South Africa. In addition, attractive evaluation results of block ciphers CLX-128 [8], TREYFER, WAKE-ROFB, DES variant, Blowfish, Spectr-H64 [9], KeeLoq [10], LED-like [11], GOST [12], and stream cipher Trivium [13], a class of hash functions [14] against slide attack have been proposed.

In almost all applications, slide attack depends heavily on the property of the subkey sequence. If subkeys adopted in each round are independent, the cipher will be secure against slide attack. In practice, however, the vast majority of block ciphers use key schedules to generate subkeys from the master keys. That is to say, the subkeys are dependent on each other. Through theory and experiment of the different types of key schedules, it has been proved that the more complex the key schedule is, the stronger is the security of ciphers against statistical attacks, owing to faster diffusion. Simple key schedule is suitable for lightweight block ciphers, which is the indispensable choice for practical applications such as sensor networks and RFID. While, if the simple key schedule is designed with not enough care, the cipher is tending to face the threat of related key attack and slide attack. What is more, it should be emphasized that complex key schedule does not necessarily mean resistant to related key attack and slide attack. Key schedule of FESH is also quite complex, while our research will show that, the complex key schedule still leads to slide property and slide attack of full round FESH.

Organization. This paper first presents the description of block cipher FESH, defines and proves its similarity properties in Section 2. Extended slide properties, and slide attacks of full round FESH are proposed in Section 3. Furthermore, Section 4 applies the extended slide properties to classical slide attacks of block cipher structures Feistel and SPN, and general frameworks are constructed. Section 5 summarizes the paper.

2. Similarity Property of FESH

2.1. Description of FESH. FESH adopts SPN structure, and the round function shown by Figure 1 can be implemented in bit slice way. The cipher is denoted as FESH-n-m while the block size is n and the key size is m. FESH-128-128, -128-192, 128-256, -256-256, -256-384, and -256-512 are supported. Corresponding round number N will be 16, 20, 20, 20, 24, and 24. This paper focuses on the most conventional versions FESH-128-128, −128-256, and −256-256.

FESH encryption firstly XORs the white key RK_0 to the plaintext P, then the result goes into N iterations. Each iteration consists of the nonlinear layer *SubNibble*, the linear layer *MixWord* and the *SubkeyAdd*.

Parallelized 32 4-bit Sboxes constitute the nonlinear layer. The n-bit state X is divided into four n/4-bit words (x_0, x_1, x_2, x_3). The i-th bit of each word is combined as a nibble $s_i = (x_0^i, x_1^i, x_2^i, x_3^i)$. s_i is the input of the i-th Sbox. The output of nonlinear layer is denoted as $Y = SubNibble(X)$. The 4-bit Sbox is shown by Table 1.

The linear layer is word-oriented. While the block size is 128 or 256-bit, the word size will be 32 or 64-bit, respectively. State Y after *SubNibble* is the input, and the output is $Z = (z_0, z_1, z_2, z_3) = MixWord(Y)$. The cyclic shift parameters are listed in Table 2.

SubkeyAdd XORs n-bit subkey to state Z.

The decryption is the inverse of the encryption, and the description is omitted here.

Denote the FESH key schedule round transformation as F. F is basically a simplified version of the encryption round function.

For FESH-128-128 and FESH-256-256:

Subkeys RK_i $(i = 0, 1, \ldots, N)$ are generated by F from the master key K.

$$\begin{aligned} RK_0 &= K \\ RK_i &= F(RK_{i-1}, Cst_{i-1}), i = 1, \ldots, N. \end{aligned} \tag{1}$$

For FESH-128-256:

Divide the $2n$-bit master key K into two n-bit RK_0 and RK_1, subkeys are generated by F from the master key K.

$$\begin{aligned} RK_0 &= K[0 \sim n-1] \\ RK_1 &= K[n \sim 2n-1] \\ RK_{i+1} &= F(RK_i, Cst_{i-1}) \oplus RK_{i-1}, i = 1, \ldots, N-1. \end{aligned} \tag{2}$$

Details of F are shown in Figure 2, including *CstAdd*, *SubNibble_K*, and *MixWord_K*.

The XORed constants, which are different in each round, destroy the symmetry of the key schedule. Each version of FESH uses different values as the initial constants Cst_0, and the rest of the constants are generated through cyclic shift of

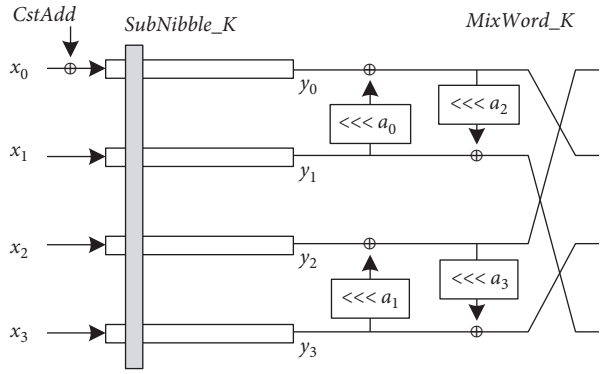

FIGURE 2: Transformation F.

TABLE 3: Parameters in *MixWord_K*.

Cipher	a_0	a_1	a_2	a_3
FESH-128-128/256	24	30	7	18
FESH-256-256	1	18	50	24

Cst_0 as $Cst_i = Cst_0 \lll i$, $i = 1, \ldots, N-1$ *or* $N-2$. Nonlinear layer *SubNibble_K* is the same as the encryption, except for different 4-bit Sbox. The linear layer selects simplified version of that of the encryption, and parameters of *MixWord_K* are shown in Table 3.

Other details of FESH are referred to the design paper [1].

2.2. Similarity Property of FESH. Through careful observations of the encryption and key schedule of FESH, following characteristics can be concluded.

(1) Both are SPN structured.
(2) Both apply a same 4-bit Sbox to each 4-bit nibble.
(3) Both *MixWord*s handle the cyclic shift and XOR operations in word.
(4) Constants are generated from an initial constant through sequential incremental cyclic shift.

Denote the encryption round function as G. Start with the input being the XOR of the state and the subkey, G satisfies similarity property as we defined in Definition 1.

Definition 1. Let X and Z be the input and output of function G, respectively, $G(X) = Z$. Then $G(X \lll_t i) = Z \lll_t i$, $i = 0, 1, \ldots, t-1$. t is 32 for FESH-128 and 64 for FESH-256. The interchangeability of G and cyclic shift is defined as similarity property of G.

Proof. For *SubNibble*, the same 4-bit Sbox is applied to each 4-bit nibble, formed by 4 bits in the same position of words x_0, x_1, x_2, x_3. It is obvious that: □

$$\begin{aligned} &SubNibble(x_0 \lll_t i, x_1 \lll_t i, x_2 \lll_t i, x_3 \lll_t i) \\ &\quad = SubNibble(x_0, x_1, x_2, x_3) \lll_t i, \end{aligned} \tag{3}$$

i.e., $SubNibble\ (X \lll_t i) = Y \lll_t i$.

MixWord is constructed by the cyclic shift and XOR operations in word,

$$\begin{aligned} &MixWord(y_0 \lll_t i, y_1 \lll_t i, y_2 \lll_t i, y_3 \lll_t i) \\ &\quad = MixWord(y_0, y_1, y_2, y_3) \lll_t i, \end{aligned} \tag{4}$$

i.e., $MixWord\ (Y \lll_t i) = Z \lll_t i$.

Function G, as a concatenation of *SubNibble* and *MixWord*, satisfies $G(X \lll_t i) = Z \lll_t i$, i.e., similarity property.

F, as the round transformation of FESH key schedule, start with the input being the XOR of the key state and the constant. In view of its similar structure with G, the following similarity property of F can be obtained directly.

Property 1. *Let W and U be the input and output of transformation F, respectively, $F(W) = U$. Then $(W \lll_t i) = U \lll_t i, i = 0, 1, \ldots, t-1$. t is 32 for FESH-128 and 64 for FESH-256.*

3. Slide Attack of FESH

3.1. Slide Key Pair. Based on Property 1, a so-called slide key pair can be constructed and defined, subkey sequences generated from the slide key pair satisfy the relation of ordered cyclic shift equality.

Definition 2. For FESH-128-128, K is the 128-bit master key, and the subkey sequence is as follows:

$$\begin{aligned} &RK_0 = K \\ &RK_1 = F(RK_0 \oplus c_0) \\ &\ldots\ldots \\ &RK_{16} = F(RK_{15} \oplus c_{15}). \end{aligned} \tag{5}$$

In which,

$$\begin{aligned} &c_0 = (cst_0, 0, 0, 0) \\ &c_1 = (cst_0 \lll_{32} 1, 0, 0, 0) = c_0 \lll_{32} 1 \\ &\ldots\ldots \\ &c_{15} = c_{14} \lll_{32} 1 = c_0 \lll_{32} 15. \end{aligned} \tag{6}$$

Define $K^* = F^{-1}(RK_0 \lll_{32} 1) \oplus c_0$, the corresponding subkey sequence of K^* satisfies $SK_j = RK_{j-1} \lll_{32} 1, j \geqq 1$. (K, K^*) is defined as a slide key pair for FESH-128-128.

Proof. For master key K, choose the related master key as $K^* = F^{-1}(RK_0 \lll_{32} 1) \oplus c_0$. □

Subkeys corresponding to K^* is:

$$\begin{aligned}
SK_0 &= K^* = F^{-1}(RK_0 \lll_{32} 1) \oplus c_0 \\
SK_1 &= F(SK_0 \oplus c_0) = F(F^{-1}(RK_0 \lll_{32} 1) \oplus c_0 \oplus c_0) \\
&= RK_0 \lll_{32} 1 \\
SK_2 &= F(SK_1 \oplus c_1) = F(RK_0 \lll_{32} 1 \oplus c_0 \lll_{32} 1) \\
&= F(RK_0 \oplus c_0) \lll_{32} 1 \\
&= RK_1 \lll_{32} 1 \\
SK_3 &= F(SK_2 \oplus c_2) = F(RK_1 \lll_{32} 1 \oplus c_1 \lll_{32} 1) \\
&= F(RK_1 \oplus c_1) \lll_{32} 1 \\
&= RK_2 \lll_{32} 1 \\
&\cdots\cdots \\
SK_{16} &= RK_{15} \lll_{32} 1.
\end{aligned} \tag{7}$$

The proof is based on the similarity property of F. The attacker obtained (K, K^*), and except for the white key SK_0 of K^*, the subkey sequences corresponding to the slide key pair constitute a cyclic shift-slide property of ordered dislocation.

For FESH-256-256, slide key pair can be defined and constructed in a similar way as FESH-128-128. The only difference is the substitute of $\lll_{32}$ by $\lll_{64}$.

Definition 3. For FESH-256-256, K is the 256-bit master key, and the subkey sequence is:

$$\begin{aligned}
RK_0 &= K \\
RK_1 &= F(RK_0 \oplus c_0) \\
&\cdots\cdots \\
RK_{24} &= F(RK_{23} \oplus c_{23}).
\end{aligned} \tag{8}$$

In which,

$$\begin{aligned}
c_0 &= (cst_0, 0, 0, 0) \\
c_1 &= (cst_0 \lll_{64} 1, 0, 0, 0) = c_0 \lll_{64} 1 \\
&\cdots\cdots \\
c_{23} &= c_{22} \lll_{64} 1 = c_0 \lll_{64} 23.
\end{aligned} \tag{9}$$

Define $K^* = F^{-1}(RK_0 \lll_{64} 1) \oplus c_0$, the corresponding subkey sequence $SK_0 (= K^*), SK_1, SK_2, \ldots, SK_{24}$ satisfies $SK_j = RK_{j-1} \lll_{64} 1, j \geqq 1$. (K, K^*) is defined as a slide key pair for FESH-256-256.

Later in this paper, without specific notification, $\lll$ will be used to refer to $\lll_{32}$ for simplification.

Key schedule of FESH-128-256 is slightly different. However, the word-based cyclic shift is not prevented. Based on Property 1, a slide key pair can also be constructed, while corresponding subkey sequences satisfy the cyclic shift-slide property of the ordered dislocation.

Definition 4. For FESH-128-256, $K = RK_0 \| RK_1$ is the 256-bit master key, and the subkey sequence is as follows:

$$\begin{aligned}
&RK_0 \\
&RK_1 \\
&RK_2 = F(RK_1 \oplus c_0) \oplus RK_0 \\
&\cdots\cdots \\
&RK_{20} = F(RK_{19} \oplus c_{18}) \oplus RK_{18}.
\end{aligned} \tag{10}$$

In which,

$$\begin{aligned}
c_0 &= (cst_0, 0, 0, 0) \\
c_1 &= (cst_0 \lll 1, 0, 0, 0) = c_0 \lll 1 \\
&\cdots\cdots \\
c_{18} &= c_{17} \lll 1 = c_0 \lll 18.
\end{aligned} \tag{11}$$

Define $K^* = F(RK_0 \lll 1 \oplus c_0) \oplus (RK_1 \lll 1) \| (RK_0 \lll 1)$, $SK_0, SK_1, SK_2, \ldots, SK_{20}$ satisfies $SK_j = RK_{j-1} \lll 1, j \geqq 1$. (K, K^*) is defined as a slide key pair for FESH-128-256.

Proof.

$$\begin{aligned}
SK_0 &= F(RK_0 \lll 1 \oplus c_0) \oplus (RK_1 \lll 1) \\
SK_1 &= RK_0 \lll 1 \\
SK_2 &= F(SK_1 \oplus c_0) \oplus SK_0 \\
&= F(RK_0 \lll 1 \oplus c_0) \oplus F(RK_0 \lll 1 \oplus c_0) \oplus (RK_1 \lll 1) \\
&= RK_1 \lll 1 \\
SK_3 &= F(SK_2 \oplus c_1) \oplus SK_1 \\
&= F(RK_1 \lll 1 \oplus c_1) \oplus RK_0 \lll 1 \\
&= RK_2 \lll 1 \\
&\cdots\cdots \\
SK_{20} &= RK_{19} \lll 1.
\end{aligned} \tag{12}$$

□

Slide key pair of FESH-128-256 is constructed, and except for the white key SK_0 of K^*, the two subkey sequences corresponding to the pair satisfy cyclic shift-slide property.

3.2. Extended Slide Property and Slide Attack of FESH. It has been introduced that, similarity properties of key schedule of FESH can be used to construct slide key pair, so that the corresponding subkey sequences constitute cyclic shift-slide property. Reviewing similarity property of the encryption G, its combination with the key transformation will yield a slide property of FESH. Since this property does not fall into traditional equal slide property, it will be called as extended slide property in our work.

3.2.1. Analysis of FESH-128-128. As proved, for slide key pair (K, K^*), except for the white key of K^*, the corresponding subkey sequences exist cyclic shift property of ordered dislocation. Extended slide property of FESH-128-128 is proposed as Property 2.

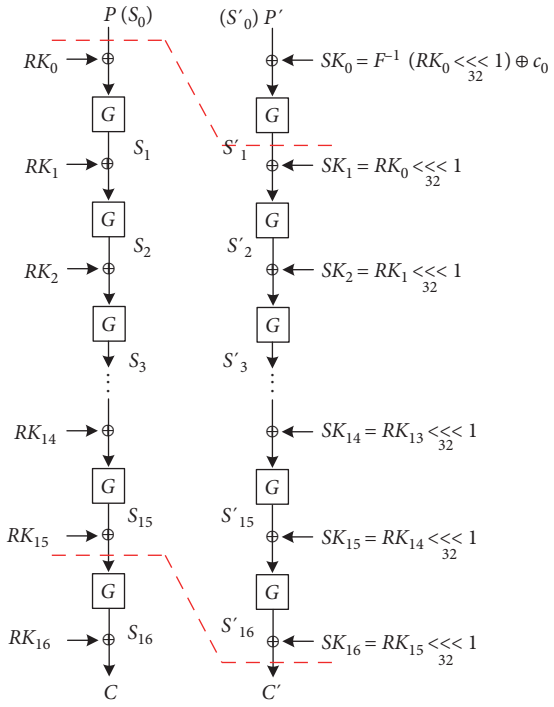

FIGURE 3: Extended slide property of FESH-128-128.

Property 2. *Suppose* $K^* = F^{-1}(RK_0 \lll 1) \oplus c_0$. *Given* $G(P' \oplus SK_0) = P \lll 1$, *then* $C' = G^{-1}(C \oplus RK_{16}) \lll 1$. *Similarly, given* $C' = G^{-1}(C \oplus RK_{16}) \lll 1$, *then* $G(P' \oplus SK_0) = P \lll 1$.

Under the premise of slide key pair, the establishment of Property 2 is due to the similarity property of G.

As in Figure 3, given $G(X \lll_t i) = Z \lll_t i$, $S'_1 = G(P' \oplus SK_0) = S_0 \lll 1$. Then, $S'_2 = G(S'_1 \oplus SK_1) = G(P \lll 1 \oplus RK_0 \lll 1) = G(P \oplus RK_0) \lll 1 = S_1 \lll 1$.

Successively, $S'_{16} = S_{15} \lll 1$, namely $C' = G^{-1}(C \oplus RK_{16}) \lll 1$.

Now, a framework similar to classical slide property is constructed. The differences are: (1) FESH is SPN, not Feistel structure; (2) the two subkey sequences have slide property of cyclic shift, with the subkeys of the beginning and ending round sharing no simple relation, besides the key schedule. While, subkey sequences of classical slide attack usually have same subkeys or a short period.

The slide attack procedures of FESH-128-128 are:

(1) Choose 2^{64} ciphertexts C and 2^{64} ciphertexts C'. For 2^{128} ciphertext pairs (C, C'), one pair satisfying $C' = G^{-1}(C \oplus RK_{16}) \lll 1$ is expected.

(2) For each ciphertext pair, compute RK_{16} through $C' = G^{-1}(C \oplus RK_{16}) \lll 1$. Further the master key K is computed from RK_{16} and then SK_0 from K. With 2^{65} queries of the decryption oracle, triplet (P, P', SK_0) will be obtained.

(3) Judge (P, P', SK_0) through $G(P' \oplus SK_0) = P \lll 1$. Only one triplet is expected, namely only one candidate of the master key can pass the test.

(4) Choose a plaintext–ciphertext pair, and verify the correctness of the candidate master key.

The data complexity is 2^{65} chosen ciphertexts. For each pair, Steps (2) and (3) can be calculated and verified in real time, and there will be no memory requirement. Next, combined with the cost ratio of reduced round encryption to full round encryption, the time complexity will be evaluated with full round encryption as the calculation unit.

When the key schedule is simple enough, its cost is negligible compared with the cost of encryption. This situation applies to the most commonly used way in lightweight designs at present.

There are 2^{128} ciphertext pairs to deal with in Step (2). RK_{16} will be computed through $C' = G^{-1}(C \oplus RK_{16}) \lll 1$, with only one G operation involved. The complexity can be calculated as $2^{128} \times \frac{1}{16} = 2^{124}$ FESH encryptions. There are 2^{128} triplets (P, P', SK_0) to be verified through $G(P' \oplus SK_0) = P \lll 1$ in Step (3), also one G is involved in each test. The time complexity can be calculated as $2^{128} \times \frac{1}{16} = 2^{124}$ FESH encryptions. To sum up, the overall time complexity is about 2^{125} FESH encryptions.

It's obvious that increasing the round number of FESH will not strengthen its security against slide attack, while in the contrary, will reduce the complexity, owing to the reduction of the cost ratio of reduced round encryption to full round encryption. This is quite different from the traditional recognition of security of iterative block ciphers.

As a supplement, evaluation of the complexity, considering the cost of key schedule will be presented in the appendix. The complexity is still better than exhaustive search.

Slide attack of FESH-256-256 is basically the same as the slide attack of FESH-128-128, with different word size. Corresponding introduction will be omitted in this paper.

3.2.2. Analysis of FESH-128-256. For slide key pair (K, K^*), except for the white key of K^*, the corresponding subkey sequences satisfy $SK_j = RK_{j-1} \lll 1, j \geqq 1$.

Property 3. *Suppose* $K^* = F(RK_0 \lll 1 \oplus c_0) \oplus (RK_1 \lll 1) || (RK_0 \lll 1)$. *Given* $G(P' \oplus SK_0) = P \lll 1$, *then* $C' = G^{-1}(C \oplus RK_{20}) \lll 1$. *Similarly, given* $C' = G^{-1}(C \oplus RK_{20}) \lll 1$, *then* $G(P' \oplus SK_0) = P \lll 1$.

As shown by Figure 4, the derivation of Property 3 is similar to that of Property 2.

Since chosen ciphertext attack is provided for FESH-128-128, chosen plaintext attack will be adopted to FESH-128-256. The attack procedures are:

(1) Choose 2^{64} plaintexts P and 2^{64} plaintexts P'. For 2^{128} plaintext pairs (P, P'), one pair is expected to satisfy $G(P' \oplus SK_0) = P \lll 1$.

(2) For each plaintext pair, compute SK_0 through $G(P' \oplus SK_0) = P \lll 1$. Guess SK_1 and there will be 2^{256} candidates for master key K. Further RK_{20} can be calculated from K and triplet (C, C', RK_{20}) is obtained.

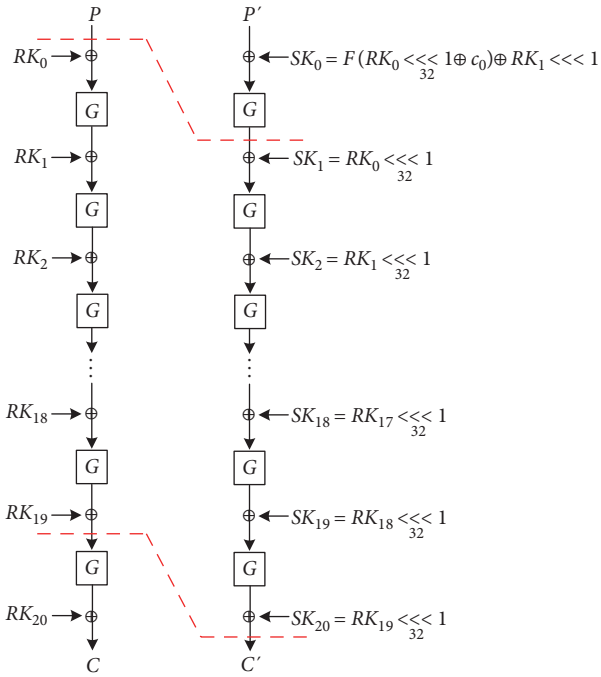

FIGURE 4: Extended slide property of FESH-128-256.

(3) Judge the triplet (C, C', RK_{20}) through $C' = G^{-1}(C \oplus RK_{20}) \lll 1$. About 2^{128} triplets are expected to pass the test, namely 2^{128} candidates of the master key.

(4) Choose two plaintext–ciphertext pairs, and verify the correctness of the candidate keys.

The data complexity is 2^{65} chosen plaintexts.

There are 2^{128} ciphertext pairs to deal with for each candidate of SK_1 in Step (2). SK_0 can be computed through $G(P' \oplus SK_0) = P \lll 1$, with only one G operation involved, and the complexity can be calculated as $2^{256} \times \frac{1}{20} = 2^{251.68}$ FESH encryptions. There are 2^{256} triplets (C, C', RK_{20}) to be verified through $C' = G^{-1}(C \oplus RK_{20}) \lll 1$ in Step (3), and only one G involved in each verification. The time complexity can be calculated as $2^{256} \times \frac{1}{20} = 2^{251.68}$ FESH encryptions. To sum up, the overall time complexity is about $2^{252.68}$ FESH encryptions.

Review the attack of FESH, the framework is similar to general slide attack. The differences are that, the relations between subkeys and intermediate states are not sliding equality. For general ciphers, similarity properties of subkey sequence usually may not be effectively used. However, if the encryption part satisfies the similarity property in the meantime, extended slide property of the cipher will be possible. *As the above research of FESH, the r-round relationship between plaintext, ciphertext and key is reduced to one-round relationship between two plaintext–ciphertext pairs and two keys.*

4. Extension to Classical Structures

Apply the idea to block cipher structures. The main point is the extension from cyclic shift property to some general similarity properties. Without loss of generality, symbol "$\trianglelefteq$" is adopted to represent the relation of similarity. In this section, assume there is a similarity property "$\trianglelefteq$" consistent with encryption and the key schedule. General framework for Feistel and SPN structures can be constructed.

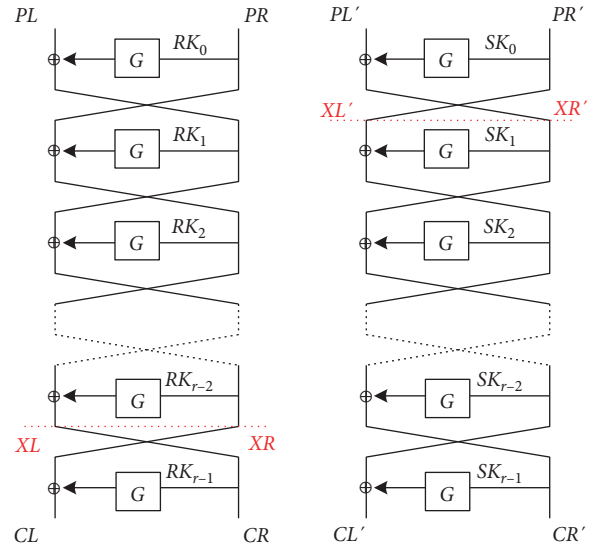

FIGURE 5: Framework for Feistel structure.

4.1. Framework for Feistel Structure. Denote the block size as n, the key size as k, the subkey size as $n/2$, the round number as r. Without loss of generality, set the gap of sliding be one round. The framework is shown by Figure 5.

Denote one round encryption of Feistel structure as RF, and the cipher satisfies extended slide property: $RF(PL'||PR', SK_0) \trianglelefteq PL||PR \Leftrightarrow RF(CL'||CR', RK_{r-1}) \trianglelefteq CL||CR$.

The attack procedures of Feistel structure are:

(1) Choose $2^{\frac{n}{4}}$ plaintexts $P_0, P_1, \ldots, P_{2^{\frac{n}{4}}-1}$, satisfying the left $\frac{n}{2}$-bit is PL, and the right $\frac{n}{2}$-bit is arbitrary.

(2) Choose $2^{\frac{n}{4}}$ plaintexts $P'_0, P'_1, \ldots, P'_{2^{\frac{n}{4}}-1}$, satisfying the right $\frac{n}{2}$-bit $PR' = PL$, and the left $\frac{n}{2}$-bit is arbitrary.

(3) Search for plaintext pair (P_i, P'_j) satisfying $PR \trianglelefteq XR' = PL' \oplus G(PR', SK_0)$. The search condition is verifying $CL' = CR$ and the probability is $2^{-\frac{n}{2}}$.

(4) Recover SK_0. For plaintext pair satisfying $PR \trianglelefteq XR' = PL' \oplus G(PR', SK_0)$, SK_0 is the only unknown variable.

(5) Recover master key bits derivable from SK_0 according to the key schedule, and other key bits through exhaustive search.

The data complexity will be $2^{\frac{n}{4}}$ chosen plaintexts.

For time complexity, there are $2^{\frac{n}{2}}$ verifications in Step (3). Feistel structure updates only half of the state in one round. Owing to this feature, direct testing of ciphertexts can be applied, and only one candidate is expected to pass the test. Complexity of Step (4) is thus reduced or even can be ignored. In this occasion, the framework and complexity are roughly the same as classical slide attack. The difference is the extension from sliding equality to more general similarity properties.

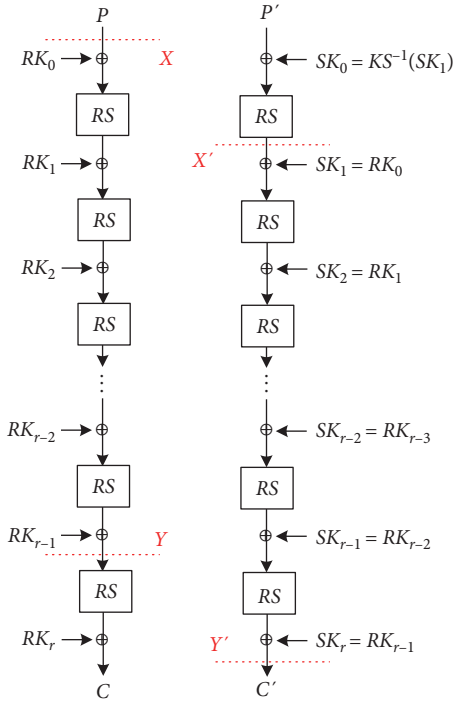

Figure 6: Framework for SPN.

4.2. Framework for SPN Structure. Targeting SPN structure, all state bits in each round will be updated, that is, there is no sieving probability in Step (3), and all pairs will pass to Step (4) for further testing. In this case, a more detailed way considering the cost ratio of partial encryption to full encryption must be adopted, just as the occasion of FESH. There are two general cases should be discussed in this section: I. Simple key schedule, which ignores the cost of key transformation; II. General key schedule, while lighter than the encryption. As introduction of the attack of FESH can serve as a demonstration for Case I, only Case II will be explored further.

As shown by Figure 6, denote one round encryption of SPN structure as *RS* and the cipher satisfies extended slide property: $RS(P', SK_0) \backsimeq P \Leftrightarrow RS(Y', RK_r) \backsimeq C$.

Focus on the accurate complexity evaluation of Case II. Set the cost of one round encryption as t and the cost of one round key transformation as t'.

The data complexity will be $2^{\frac{n}{2}}$ chosen plaintexts.

For time complexity, in Steps (2) and (3), there are two verifications for each of the 2^n pairs, which should be $2^n \times \frac{2}{r}$ r-round encryptions; calculation of RK_r from SK_0 should be $2^n \times \frac{t'}{t}$ r-round encryptions. Complexity of Steps (2) and (3) can be evaluated as $2^n \times \frac{2}{r} + 2^n \times \frac{t'}{t}$.

It should be noted that in the general framework, RK_r can be calculated by SK_0 by default. In practical analysis, it is possible that only a portion of RK_r can be calculated by SK_0, and the remaining bits of RK_r should be guessed.

5. Conclusion

This paper starts with the considerations of block cipher FESH, explore its encryption and key structures, similarity properties and extended slide attack of full round FESH are proposed. Furthermore, the idea is extended to construct more general frameworks and complexity evaluations for Feistel and SPN structures. The primary intent of this paper is to emphasize the importance for cipher designers to exercise increased caution when dealing with the key schedule and constant generation, in order to mitigate the emergence of undesirable properties.

Slide attack of Feistel or similar structures have already attracted much attention. Furthermore, it is worth emphasize that, although the complexity of slide attack of SPN structure is quite close to the complexity of exhaustive attack, it indeed also reflects vulnerability of the design. With further possible development and application of time memory tradeoff technique, collision attack, and the popularize of quantum technique, the complexity is potentially to be reduced and practical security problems will perhaps arise. Accordingly, evaluation of SPN structures against slide attack is quite research worthy.

Appendix

Complexity Reevaluated of Slide Attack of FESH-128-128

Firstly, the cost of key transformation of FESH-128-128 is converted to the cost of encryption function of FESH-128-128 according to the number of instructions.

One key transformation includes: *CstAdd*, *SubNibble_K*, and *MixWord_K*. The constant is added to one word, so one instruction is needed. *SubNibble_K* can be implemented by 9 instructions taking advantage of the bit slice technique. Each cyclic shift needs 3 instructions, and there are 4 cyclic shift operations in *MixWord_K*. In summary, $(1 + 9 + 12 + 4) \times 16 = 416$ instructions are needed for 16-round key schedule.

One encryption function includes: *SubNibble*, *MixWord*, and *SubkeyAdd*. The Sbox layer can be implemented by 15 instructions. there are 8 cyclic shift operations in *MixWord*, so 24 instructions are needed. Each *SubkeyAdd* can be implemented by 4 instructions. All in all, $4 + (4 + 15 + 24 + 8) \times 16 = 820$ instructions are needed for 16-round encryption.

Complexity to calculate RK_{16} in Step (2) will be $2^{128} \times \frac{1}{16} = 2^{124}$ 16-round encryptions; 2^{128} times of key updating are needed to calculate the master key from RK_{16}, which equals $2^{128} \times \frac{416}{820} = 2^{128-0.98}$ 16-round encryptions; one key updating is needed to calculate SK_0 from the master key, which leads to $2^{128} \times \frac{1}{16} \times 2^{-0.98} = 2^{123.02}$ 16-round encryptions; complexity of Step (3) will be $2^{128} \times \frac{1}{16} = 2^{124}$ full FESH encryptions. All in all, the time complexity of the attack will be $2^{127.27}$ FESH encryptions.

Data Availability

All data for this research are available in the paper.

Conflicts of Interest

The authors have no conflict of interest with regard to this work.

Authors' Contributions

Yafei Zheng completed the theoretical derivation and wrote the manuscript; Wenling Wu contributed significantly to the analysis and the manuscript preparation.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No. 62072445).

References

[1] K. Jia, X. Dong, C. Wei, Z. Li, H. Zhou, and T. C, "On the design of block cipher FESH," *Journal of Cryptologic Research*, vol. 6, no. 6, pp. 713–726, 2019.

[2] A. Biryukov and D. Wagner, "Slide attacks," in *FSE 1999*, L. Knudsen, Ed., vol. 1636 of *Lecture Notes in Computer Science*, pp. 245–259, Springer, Berlin, Heidelberg, 1999.

[3] E. Biham, "New types of cryptanalytic attacks using related keys (extended abstract)," in *EUROCRYPT 1993*, T. Helleseth, Ed., vol. 765 of *Lecture Notes in Computer Science*, pp. 398–409, Springer, Berlin, Heidelberg, 1993.

[4] A. Biryukov and D. Wagner, "Advanced slide attacks," in *EUROCRYPT 2000*, B. Preneel, Ed., vol. 1807 of *Lecture Notes in Computer Science*, pp. 589–606, Springer, Berlin, Heidelberg, 2000.

[5] E. Biham, O. Dunkelman, and N. Keller, "Improved slide attacks," in *FSE 2007*, A. Biryukov, Ed., vol. 4593 of *Lecture Notes in Computer Science*, pp. 153–166, Springer, Berlin, Heidelberg, 2007.

[6] NRS 009-6-7, "Rationalized User Specification, Electricity Sales Systems, Part 6: Interface standards Section 7: Standard Transfer Specifica-tion/Credit Dispensing Unit—Electricity dispenser—Token Encoding and Data Encryption and Decryption," 2002.

[7] M. Sönmez Turan, "Related key slide attacks on block ciphers with secret components," in *LightSec 2013*, vol. 8162 of *Lecture Notes in Computer Science*, pp. 28–42, Springer, Berlin, Heidelberg, 2013.

[8] A. Mege, "Slide attack on CLX-128," in *Proceedings of the Lightweight Cryptography Workshop*, 169, 2019.

[9] S. Kavut and M. D. Yücel, "Slide attack on Spectr-H64," in *INDOCRYPT 2002*, A. Menezes and P. Sarkar, Eds., vol. 2551 of *Lecture Notes in Computer Science*, pp. 34–47, Springer, Berlin, Heidel-berg, 2002.

[10] N. T. Courtois, "Self-similarity attacks on block ciphers and application to KeeLoq," in *Cryptography and Security: From Theory to Applications*, D. Naccache, Ed., vol. 6805 of *Lecture Notes in Computer Science*, pp. 55–66, Springer, Berlin, Heidelberg, 2012.

[11] L. Xu, J. Guo, J. Cui, and M. Li, "Key-recovery attacks on LED-like block ciphers," *Tsinghua Science and Technology*, vol. 24, no. 5, pp. 585–595, 2019.

[12] L. Lu and S. Chen, "A compress slide attack on the full GOST block cipher," *Information Processing Letters*, vol. 113, no. 17, pp. 634–639, 2013.

[13] A. Baksi, S. Maitra, and S. Sarkar, "An improved slide attack on Trivium," in *IPSI Transaction on Internet Research*, pp. 351–375, IPSI Transactions on Internet Research, or TIR, France, Paris, 2015.

[14] M. Gorski, S. Lucks, and T. Peyrin, "Slide attacks on a class of hash functions," in *ASIACRYPT 2008*, vol. 5350, pp. 143–160, LNCS, Springer, Heidelberg, 2008.