Security Evaluation on Simeck against Zero Correlation Linear Cryptanalysis

Kai Zhang[1,2], Jie Guan[1], Bin Hu[1], Dongdai Lin[2], Wentao Zhang[2]

[1] Information Science and Technology Institute

Zhengzhou 450000, China

[2] State Key Laboratory of Information Security, Institute of Information Engineering,

Chinese Academy of Sciences, Beijing 100093, China

[e-mail: zhkai2010@139.com, guanjie007@163.com, hb2110@126.com, ddlin@iie.ac.cn, zhangwentao@iie.com]

Abstract: SIMON and SPECK family ciphers have attracted the attention of cryptographers all over the world since proposed by NSA in June, 2013. At CHES 2015, Simeck, a new block cipher inspired from both SIMON and SPECK is proposed, which is more compact and efficient. However, the security evaluation on Simeck against zero correlation linear cryptanalysis seems missing from the specification. The main focus of this paper is to fill this gap and evaluate the security level on Simeck against zero correlation linear cryptanalysis. According to our study, 11/13/15 rounds zero correlation linear distinguishers on Simeck32/48/64 are proposed respectively, then zero correlation linear cryptanalysis on 20/24/27 rounds Simeck32/48/64 are firstly proposed. As far as we know, for Simeck32, our result is the best result to date.

Key Words: Cryptanalysis; Lightweight Block Cipher; Zero Correlation Linear Cryptanalysis; Simeck

1 Introduction

With the increasing need for low-end embedded devices, algorithms which have to adapt to the very constrained memory is greatly needed. SIMON and SPECK[4] are two lightweight block ciphers proposed by NSA in 2013. SIMON is designed for hardware friendly and SPECK is designed for software friendly. To meet the need of small hardware implementations, Gangqiang Yang et al. proposed Simeck at CHES 2015. Simeck benefits from both the designs of SIMON and SPECK, while it has more compact and efficient hardware implementation. At the same time, it has comparable security margin with SIMON and SPECK.

The concept of zero correlation linear cryptanalysis was firstly proposed by Andrey Bogdanov and Vincent Rijmen in [5]. In the recent two years, zero correlation linear cryptanalysis has shown its great potential in cryptanalysis and it has proven to be effective against massive ciphers. Generally speaking, this cryptanalytic method can be concluded as “use linear approximation of probability 1/2 to eliminate the wrong key candidates”.

In the specification of Simeck [1], designers evaluated the security level against massive cryptanalytic methods such as differential cryptanalysis, linear cryptanalysis, impossible differential cryptanalysis, rotational attacks and so on. However, as an important cryptanalytic tool, zero correlation linear cryptanalysis is essential to evaluate the security level for this

* This work was supported by the National Natural Science Foundation of China under Grant No. 61202491, 61272041, 61272488, 61402523 and 61572516.

newly proposed block cipher.

Our contributions

The main purpose of this paper is to evaluate the security level on Simeck against zero correlation linear cryptanalysis.

- For distinguishers, this paper proposes zero correlation linear distinguishers on all variants of Simeck. Based on "0-1" bit contradictions, 11/13/15-round zero correlation linear distinguishers on Simeck32/48/64 are firstly proposed.
- To validate the usefulness of these newly proposed zero correlation linear distinguishers, this paper evaluates the security level on Simeck against zero correlation linear cryptanalysis, for Simeck32/48/64, 20/24/27 round of the cipher can be attacked respectively. The results are concluded in the table 1 below:

Table1. Summary of all the cryptanalysis on Simeck

Algorithm	Attack	Rounds Attacked	Data	Memory	Time	Reference
Simeck32(64)	DC	19	$O(2^{31.5})$	--	2^{34}	[1]
	DC	19	$O(2^{31})$	2^{31}	2^{31}	[3]
	LC	$18^{\clubsuit}$	$O(2^{32})$	--	$2^{62.56}$	[2]
	ID	20	$O(2^{32})$	2^{56}	$2^{62.6}$	[1]
	ZC	**20**	$\boldsymbol{O(2^{32})}$	$\mathbf{2^{39.37}}$	$\mathbf{2^{56.65}}$	**this paper**
Simeck48(96)	DC	20	$O(2^{46})$	--	2^{75}	[1]
	DC	26	$O(2^{47})$	2^{47}	2^{62}	[3]
	LC	19	$O(2^{45})$	--	2^{94}	[2]
	ID	24	$O(2^{48})$	2^{74}	$2^{94.7}$	[1]
	ZC	**24**	$\boldsymbol{O(2^{48})}$	$\mathbf{2^{65.06}}$	$\mathbf{2^{91.6}}$	**this paper**
Simeck64(128)	DC	26	$O(2^{63})$	--	2^{121}	[1]
	DC	33	$O(2^{63})$	2^{63}	2^{96}	[3]
	LC	$27^{\clubsuit}$	$O(2^{61})$	--	$2^{120.5}$	[2]
	ID	25	$O(2^{64})$	2^{79}	$2^{126.6}$	[1]
	ZC	**27**	$\boldsymbol{O(2^{64})}$	$\mathbf{2^{74.34}}$	$\mathbf{2^{112.79}}$	**this paper**

"$^{\clubsuit}$": The success probability of the attack is 0.477.

DC: differential cryptanalysis; LC: linear cryptanalysis; ID: impossible differential cryptanalysis; ZC: zero correlation linear cryptanalysis.

This paper is organized as follows. In section 2, notations used in this paper are introduced. In section 3, a brief description on Simeck is presented. Section 4 proposes some zero correlation linear distinguishers on Simeck. Section 5 proposes zero correlation linear cryptanalysis on Simeck and section 6 concludes this paper.

2 Notations

- Simeck $2n(4n)$: the block size of this Simeck version is $2n$, the key size is $4n$.
- P: a $2n$-bit plaintext, consisting of two n-bit words $P=(P_L,P_R)$;
- C: a $2n$-bit ciphertext, consisting of two n-bit words $C=(C_L,C_R)$;
- X^i: a $2n$-bit intermediate value (input of i-th round in the encryption), consisting of two n-bit words $X^i=(X_L^i,X_R^i)$, the jth bit of X_L^i,X_R^i are denoted as $X_L^i(j)$ and $X_R^i(j)$ respectively;

- K: master key. For Simeck $2n(4n)$, $K=(K_3,K_2,K_1,K_0)$;
- RK^i: an n-bit round key for the i-th round;
- $\oplus$: XOR operation;
- <<< r: left rotation for r bits;
- >>> r: right rotation for r bits;
- "&","·": bitwise AND operation.

3 A Brief Description on Simeck

Simeck is a family of lightweight block ciphers proposed at CHES 2015, the round function of Simeck is slightly modified from SIMON and the key schedule of Simeck adopt the idea of round function reuse like SPECK does. Like SIMON, the structure of Simeck is a typical Feistel network, the operations Simeck used are simply bitwise AND, XOR and rotation operation. The only difference for the round function between Simeck and SIMON is the rotation constant, which is (0,5,1) for Simeck and (1,8,2) for SIMON. The round function of Simeck is depicted in Figure 1 below. Unlike SIMON and SPECK, there are only 3 different versions for Simeck, the parameters of each version are illustrated in the table 2 below.

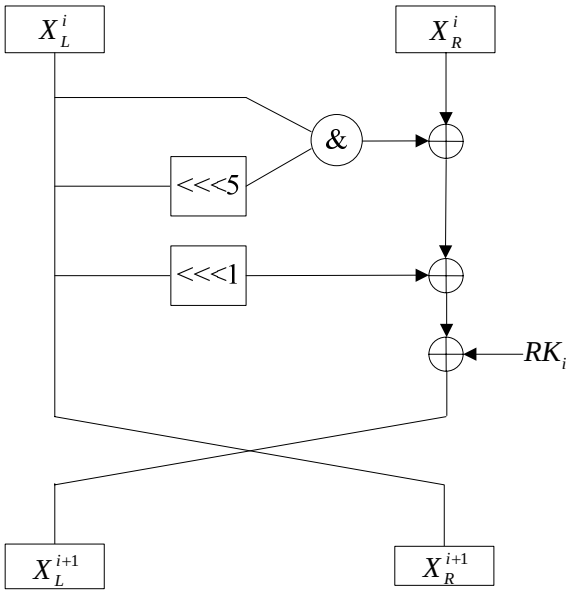

Fig1. Round function of Simeck

Table2. Simeck parameters

Algorithm	Block Size $2n$	Key Size $4n$	Word Size n	Rounds T
Simeck32(64)	32	64	16	32
Simeck48(96)	48	96	24	36
Simeck64(128)	64	128	32	44

The initial master key K consists of four n-bit words (K_3,K_2,K_1,K_0).

First, initialize vector t_0, t_1, t_2, k_0 with K_3, K_2, K_1, K_0.

$$k_0 = K_0, t_0 = K_1, t_1 = K_2, t_2 = K_3$$

For different *i*, round key $RK^i = k_i$ is computed with the following equation:

$$\begin{cases} k_{i+1} = t_i \\ t_{i+3} = k_i \oplus f(t_i) \oplus C \oplus (z_j)_i \end{cases}$$

where $f(x) = (x \cdot (x <<< 5)) \oplus (x <<< 1)$, C and $(z_j)_i$ are predefined constants. For more details of Simeck algorithm, we refer the readers to reference [1].

4 Zero Correlation Linear Distinguishers on Simeck

For ARX ciphers, zero correlation linear distinguishers are usually built with miss-in-the –middle technique and the contradiction for this kind of distinguishers is usually "0-1" contradiction. Usually, this kind of distinguishers consists of two approaches: forward approach (encryption direction) and backward approach (decryption direction).

For Simeck32, in the encryption direction, we find that for any 5-round non-zero linear hull with input linear mask of (1000000000000000 0000000000000000), the linear mask of the internal state must be (???10???00??000? ?????????0???00?). Similarly, in the decryption direction, for any 6-round non-zero linear hull with output linear mask of (0000000000000000 0000010000000000), the linear mask of the internal state must be (???0???????????? ??00??????????0?). Combining these two approaches together, we can derive an 11-round zero correlation linear distinguisher with input linear mask of (1000000000000000 0000000000000000) and output linear mask of (0000000000000000 0000010000000000). Similarly, 13-round zero correlation linear distinguisher for Simeck48 and 15-round zero correlation linear distinguisher for Simeck64 can be derived (See table 7 for more details).

The zero correlation linear distinguishers used to attack these Simeck variants are illustrated in the table 3 below.

Table3. Zero Correlation Linear Distinguishers used to Attack Simeck32/48/64

Algorithm	Position	Zero Correlation Linear Distinguisher
Simeck32 (11-round)	Input	1000000000000000 0000000000000000
	Output	0000000000000000 0000010000000000
Simeck48 (13-round)	Input	100000000000000000000000 000000000000000000000000
	Output	000000000000000000000000 000001000000000000000000
Simeck64 (15-round)	Input	10000000000000000000000000000000 00000000000000000000000000000000
	Output	00000000000000000000000000000000 00000001000000000000000000000000

5 Zero Correlation Linear Cryptanalysis on Simeck

In this section, with the proposed 11/13/15 round zero correlation linear distinguishers

on Simeck32/48/64, zero correlation linear cryptanalysis on 20-round Simeck32, 24-round Simeck48 and 27-round Simeck64 are firstly proposed.

5.1 Zero Correlation Linear Cryptanalysis on Simeck32

In this section, zero correlation linear cryptanalysis on 20-round Simeck32 is proposed, we use an 11-round zero correlation linear distinguisher and add five initial rounds and four final rounds before and after the distinguisher. The details are depicted in the figure 2 below:

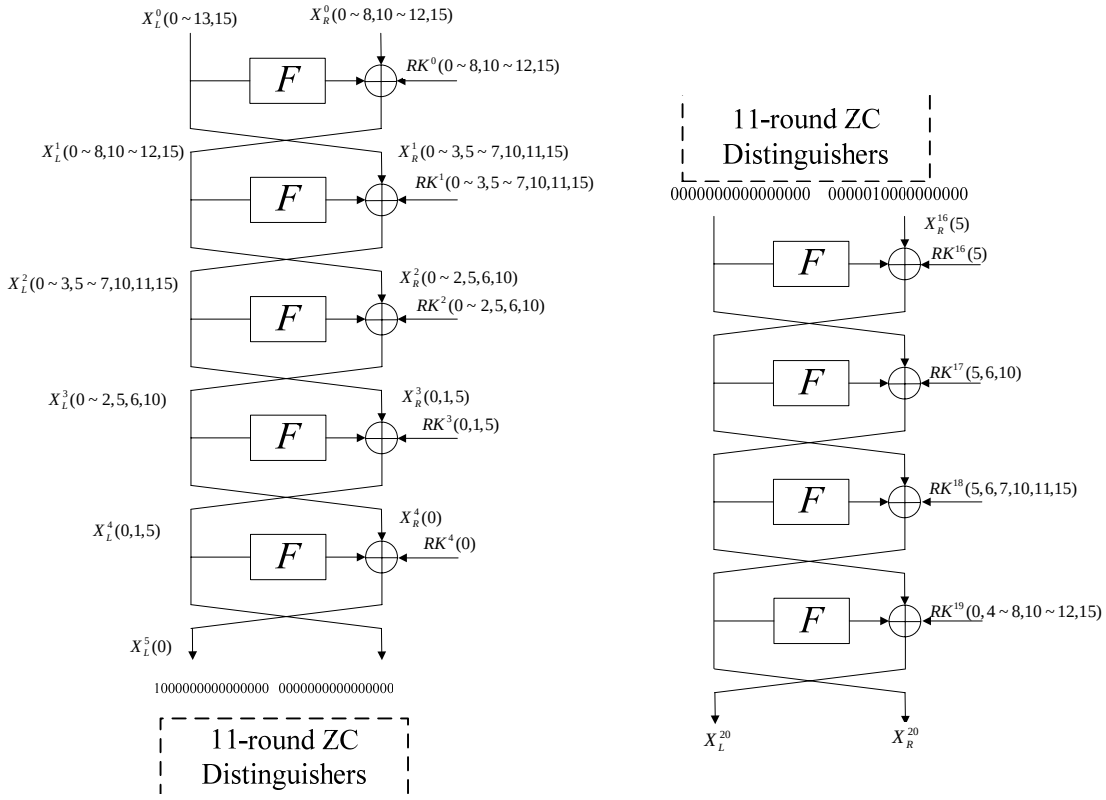

Fig2. Initial five rounds encryption (Left) and final four rounds decryption (Right)

Figure 2 just show those intermediate state values and the subkeys in the partial encryption and decryption process. With partial-sum technique, the procedure of the attack can be concluded as follows:

Step 1. Allocate a counter vector $V_1[\,X_L^0(0\sim13,15)\,|\,X_R^0(0\sim8,10\sim12,15)\,|\,X_R^{16}(5)\,]$ of size 2^{28} where each element is 8-bit length and initialized to zero;

Step 2. Guess all possible values of 20 round key bits $RK^{16}(5), RK^{17}(5,6,10)$, $RK^{18}(5,6,7,10,11,15)$, $RK^{19}(0,4\sim8,10\sim12,15)$;

Step 3. Partially decrypt the ciphertext of each (*P*, *C*) pair to get $X_R^{16}(5)$. Add one to the corresponding $V_1[\,X_L^0(0\sim13,15)\,|\,X_R^0(0\sim8,10\sim12,15)\,|\,X_R^{16}(5)\,]$;

Step 4. The target of this step is to reduce $V_1[\,X_L^0(0\sim13,15)\,|\,X_R^0(0\sim8,10\sim12,15)\,|\,X_R^{16}(5)\,]$ to a new counter vector $V_2[\,X_L^1(0\sim8,10\sim12,15)\,|\,X_R^1(0\sim3,5\sim7,10,11,15)\,|\,X_R^{16}(5)\,]$. During this step, altogether 13 bits of RK_0 have to be guessed. To reduce the time complexity, we guess these round key bits bit by bit. The guessed key, intermediate state counter and complexity are

illustrated in the table 4 below.

Table4. Details of Partial encryption procedure for step 4

Step	Guess	Counter(size)	Complexity
4.1	$RK_0(4,8)$	$V_{1.1}[\ X_L^0(0\sim 8,10\sim 13,15)\mid X_R^0(0\sim 3,5\sim 7,10\sim 12,15)\mid X_L^1(4,8)\mid X_R^{16}(5)\ ]$ (28 bits)	$2^{22}\cdot 2^{29}\cdot 2/(16\cdot 20)$
4.2	$RK_0(12)$	$V_{1.2}[\ X_L^0(0\sim 8,10\sim 12,15)\mid X_R^0(0\sim 3,5\sim 7,10,11,15)\mid X_L^1(4,8,12)\mid X_R^{16}(5)\ ]$ (27 bits)	$2^{23}\cdot 2^{28}\cdot 1/(16\cdot 20)$
4.3	$RK_0(3,15)$	$V_{1.3}[\ X_L^0(0\sim 3,5\sim 8,10\sim 12,15)\mid X_R^0(0\sim 2,5\sim 7,10,11)\mid X_L^1(3,4,8,12,15)\mid X_R^{16}(5)\ ]$ (26 bits)	$2^{25}\cdot 2^{27}\cdot 2/(16\cdot 20)$
4.4	$RK_0(7)$	$V_{1.4}[\ X_L^0(0\sim 3,5\sim 7,10\sim 12,15)\mid X_R^0(0\sim 2,5,6,10,11)\mid X_L^1(3,4,7,8,12,15)\mid X_R^{16}(5)\ ]$ (25 bits)	$2^{26}\cdot 2^{26}\cdot 1/(16\cdot 20)$
4.5	$RK_0(11)$	$V_{1.5}[\ X_L^0(0\sim 3,5\sim 7,10,11,15)\mid X_R^0(0\sim 2,5,6,10)\mid X_L^1(3,4,7,8,11,12,15)\mid X_R^{16}(5)\ ]$ (24 bits)	$2^{27}\cdot 2^{25}\cdot 1/(16\cdot 20)$
4.6	$RK_0(0\sim 2,5,6,10)$	$V_2[\ X_L^1((0\sim 8,10\sim 12,15)\mid X_R^1(0\sim 3,5\sim 7,10,11,15)\mid X_R^{16}(5)\ ]$ (24 bits)	$2^{33}\cdot 2^{24}\cdot 6/(16\cdot 20)$

Step 5. The target of this step is to reduce $V_2[\ X_L^1(0\sim 8,10\sim 12,15)\mid X_R^1(0\sim 3,5\sim 7,10,11,15)\mid X_R^{16}(5)\ ]$ to a new counter vector $V_3[\ X_L^2(0\sim 3,5\sim 7,10,11,15)\mid X_R^2(0\sim 2,5,6,10)\mid X_R^{16}(5)\ ]$. During this step, altogether 10 bits of RK_1 have to be guessed. To reduce the time complexity, we guess these round key bits bit by bit. The guessed key, intermediate state counter and complexity are illustrated in the table 5 below.

Table5. Details of Partial encryption procedure for step 5

Step	Guess	Counter(size)	Complexity
5.1	$RK_1(2,3)$	$V_{2.1}[X_L^1(0\sim 4,5\sim 8,10\sim 12,15)\mid X_R^1(0,1,5\sim 7,10,11,15)\mid X_L^2(2,3)\mid X_R^{16}(5)]$ (23 bits)	$2^{35}\cdot 2^{24}\cdot 2/(16\cdot 20)$
5.2	$RK_1(15)$	$V_{2.2}[X_L^1(0\sim 2,5\sim 8,10\sim 12,15)\mid X_R^1(0,1,5\sim 7,10,11)\mid X_L^2(2,3,15)\mid X_R^{16}(5)]$ (22 bits)	$2^{36}\cdot 2^{23}\cdot 1/(16\cdot 20)$
5.3	$RK_1(7)$	$V_{2.3}[X_L^1(0\sim 2,5\sim 7,10\sim 12,15)\mid X_R^1(0,1,5,6,10,11)\mid X_L^2(2,3,7,15)\mid X_R^{16}(5)]$ (21 bits)	$2^{37}\cdot 2^{22}\cdot 1/(16\cdot 20)$
5.4	$RK_1(6)$	$V_{2.4}[X_L^1(0\sim 2,5,6,10\sim 12,15)\mid X_R^1(0,1,5,10,11)\mid X_L^2(2,3,6,7,15)\mid X_R^{16}(5)]$ (20 bits)	$2^{38}\cdot 2^{21}\cdot 1/(16\cdot 20)$
5.5	$RK_1(11)$	$V_{2.5}[X_L^1(0\sim 2,5,6,10,11,15)\mid X_R^1(0,1,5,10)\mid X_L^2(2,3,6,7,11,15)\mid X_R^{16}(5)]$ (19 bits)	$2^{39}\cdot 2^{20}\cdot 1/(16\cdot 20)$
5.6	$RK_1(10)$	$V_{2.6}[X_L^1(0\sim 2,5,6,10)\mid X_R^1(0,1,5)\mid X_L^2(2,3,6,7,10,11,15)\mid X_R^{16}(5)]$ (17 bits)	$2^{40}\cdot 2^{19}\cdot 1/(16\cdot 20)$
5.7	$RK_1(0,1,5)$	$V_3[X_L^2(0\sim 3,5\sim 7,10,11,15)\mid X_R^2(0\sim 2,5,6,10)\mid X_R^{16}(5)]$ (17 bits)	$2^{43}\cdot 2^{17}\cdot 3/(16\cdot 20)$

Step 6. The target of this step is to reduce $V_3[X_L^2(0\sim 3,5\sim 7,10,11,15)\mid X_R^2(0\sim 2,5,6,10)\mid X_R^{16}(5)]$ to a new counter vector $V_4[X_L^3(0\sim 2,5,6,10)\mid X_R^3(0,1,5)\mid X_R^{16}(5)]$. During this step, altogether 6 bits of RK_2 have to be guessed. To reduce the time complexity, we guess these round key bits bit by bit. The guessed key, intermediate state counter and complexity are illustrated in the table 6 below.

Table6. Details of Partial encryption procedure for step 6

Step	Guess	Counter(size)	Complexity
6.1	$RK_2(2)$	$V_{3.1}[X_L^2(0\sim 2,5\sim 7,10,11,15)\mid X_R^2(0,1,5,6,10)\mid X_L^3(2)\mid X_R^{16}(5)]$	$2^{44}\cdot 2^{17}\cdot$

		(16 bits)	1/(16·20)
6.2	$RK_2(1)$	$V_{3.2}[X_L^2(0,1,5\sim 7,10,11,15) \mid X_R^2(0,5,6,10) \mid X_L^3(1,2) \mid X_R^{16}(5)]$ (15 bits)	$2^{45}\cdot 2^{16}\cdot$ 1/(16·20)
6.3	$RK_2(6)$	$V_{3.3}[X_L^2(0,1,5,6,10,11,15) \mid X_R^2(0,5,10) \mid X_L^3(1,2,6) \mid X_R^{16}(5)]$ (14 bits)	$2^{46}\cdot 2^{15}\cdot$ 1/(16·20)
6.4	$RK_2(5)$	$V_{3.4}[X_L^2(0,1,5,10,11,15) \mid X_R^2(0,10) \mid X_L^3(1,2,5,6) \mid X_R^{16}(5)]$ (13 bits)	$2^{47}\cdot 2^{14}\cdot$ 1/(16·20)
6.5	$RK_2(10)$	$V_{3.5}[X_L^2(0,1,5) \mid X_R^2(0) \mid X_L^3(1,2,5,6,10) \mid X_R^{16}(5)]$ (10 bits)	$2^{48}\cdot 2^{13}\cdot$ 1/(16·20)
6.6	$RK_2(0)$	$V_4[X_L^3(0\sim 2,5,6,10) \mid X_R^3(0,1,5) \mid X_R^{16}(5)]$ (10 bits)	$2^{49}\cdot 2^{10}\cdot$ 1/(16·20)

Step 7. The target of this step is to reduce $V_4[X_L^3(0\sim 2,5,6,10) \mid X_R^3(0,1,5) \mid X_R^{16}(5)]$ to a new counter vector $V_5[X_L^4(0,1,5) \mid X_R^4(0) \mid X_R^{16}(5)]$. During this step, altogether 3 bits of RK_3 have to be guessed. Similarly, we guess these round key bits bit by bit. The guessed key, intermediate state counter and complexity are illustrated in the table 7 below.

Table7. Details of Partial encryption procedure for step 7

Step	Guess	Counter(size)	Complexity
7.1	$RK_3(1)$	$V_4[X_L^3(0,1,5,6,10) \mid X_R^3(0,5) \mid X_L^4(1) \mid X_R^{16}(5)]$ (9 bits)	$2^{50}\cdot 2^{10}\cdot$ 1/(16·20)
7.2	$RK_3(0)$	$V_4[X_L^3(0,5,6,10) \mid X_R^3(5) \mid X_L^4(0,1) \mid X_R^{16}(5)]$ (8 bits)	$2^{51}\cdot 2^{9}\cdot$ 1/(16·20)
7.7	$RK_3(5)$	$V_5[X_L^4(0,1,5) \mid X_R^4(0) \mid X_R^{16}(5)]$ (5 bits)	$2^{52}\cdot 2^{8}\cdot$ 1/(16·20)

Step 8. Allocate a new counter vector $V_6[X_L^5(0) \mid X_R^{16}(5)]$ of size 2^2 where each element is 32-bit length and initialized to zero. As three bits $RK_1(0,1,5)$ have already been guessed, $RK_4(0)$ can be directly deduced according to these three bits. Then we can compute $X_L^5(0)$ and add one to the corresponding $V_6[X_L^5(0) \mid X_R^{16}(5)]$.

In the first 8 steps, there are altogether 52 round key bits guessed. If a round key candidate is the correct key, $V_6[0 \mid 0]+V_6[1 \mid 1]=2^{31}$ whereas for a wrong key candidate, the probability is $P(V_6[0 \mid 0]+V_6[1 \mid 1]=2^{31})=2^{(4-32)/2}/\sqrt{2\pi}\approx 2^{-15.33}$. So after the 8 steps above, the 2^{52} round key candidates can be reduced to $2^{36.67}$ approximately, then store these round key candidates.

Next step is to recover the master key. Unlike SIMON, the key schedule of Simeck is nonlinear, which lead to the idea of establishing linear equations for the round key bits and then using Gaussian elimination method to derive master keys is impractical. So we introduce a novel method here to recover the master keys.

For the 52 round key bits (denoted as $(\eta \mid \xi)$), there are altogether 32 master key bits (η) and other 20 round key bits (ξ). As for the 52 round key bits, there are $2^{36.67}$ round key candidates left, it can be regarded as 15.33 bits information have been derived. For η and ξ, 9.43 bits and 5.90 bits information have been derived respectively.

In average, among these 52 round key bits, there are $2^{32}\cdot 2^{-9.43}=2^{22.57}$ master key bits and $2^{20}\cdot 2^{-5.90}=2^{14.10}$ other round key bits left.

Step 9. First of all, sort the $2^{36.67}$ round key candidates $(\eta|\xi)$ according to the value of η.

Step 10. Then, for each candidate of η, guess all the other 32 master key bits (denoted as η'). After the key schedule algorithm, if a candidate $(\eta|\eta')$ can make $(\eta|\xi)$ locate in the $2^{36.67}$ round key candidates, save $(\eta|\eta')$.

Step 11. Finally test whether a derived master key $(\eta|\eta')$ is correct or not by verification for plaintext-ciphertext pairs.

Complexity estimation

The data complexity is $O(2^{32})$, while the memory complexity is $2^{28}\cdot 8/8+2^{36.67}\cdot 52/8 \approx 2^{39.37}$ bytes.

As for the time complexity, the time complexity for each step is as follows:

Step1-Step3: $2^{32}\cdot 2^{20}\cdot 20/(16\cdot 20) \approx 2^{48}$ 20-round Simeck32/64 encryptions.

Step4: $2^{51.30}$ 20-round Simeck32/64 encryptions (For details see table 4).

Step5: $2^{54.38}$ 20-round Simeck32/64 encryptions (For details see table 5).

Step6: $2^{55.07}$ 20-round Simeck32/64 encryptions (For details see table 6).

Step7: $2^{53.26}$ 20-round Simeck32/64 encryptions (For details see table 7).

Step8: $2^{52}\cdot 2^{5}\cdot 1/(16\cdot 20) \approx 2^{48.67}$ 20-round Simeck32/64 encryptions.

Step9-Step10: For each η, there are altogether 2^{32} corresponding η' needs to be guessed. As there are $2^{22.57}$ η, so the total time complexity for this step is $2^{32}\cdot 2^{22.57}=2^{54.57}$ 20-round Simeck32/64 key schedule algorithm. Ideally, there are $2^{32}\cdot 2^{-5.90}=2^{26.10}\eta'$ left for each η. So the number of master key candidates $(\eta|\eta')$ left is $2^{26.10}\cdot 2^{22.57}\approx 2^{48.67}$.

Step11: Test the correctness of each master key candidate $(\eta|\eta')$ with plaintext-ciphertext pairs (P, C). For the correct master key κ, after encryption, $E_\kappa(P)=C$ with probability one. For a false guess, after encryption, $\text{Prob}(E_\kappa(P)=C)=1/2^{32}$. So after sieving with first plaintext-cipher pair, the space of master key candidate can be reduced to about $2^{16.67}$, these master key candidates should test through the second plaintext-ciphertext pair. Iterate this process until the only correct key is left. The time complexity for this step is about $2^{48.67}$ 20-round Simeck32/64 encryptions.

To sum up, the total time complexity is about $2^{56.65}$ 20-round Simeck32/64 encryptions.

5.2 Zero Correlation Linear Cryptanalysis on Simeck48 and Simeck64

For Simeck48 and Simeck64, similar attacks are also possible. However, to avoid redundancy, this section just uses two small tables (table 8 and table 9) to illustrate the counter vector, the order of the guessed key bits and complexities of each and overall procedure. Researchers could follow these details to realize the procedure of the attack. Specially, we want to note that the round key bits described in these tables have orders, to decrease the overall complexity, we adopt the strategy to guess the round key bits bit by bit (despite the first steps), those key bits in the bracket are guessed at one time.

Table8. Attack procedure on 24-round Simeck48(96)

Step	Guess	Counter	Complexity
1	$RK^{19}(5), RK^{20}(5,6,10),$ $RK^{21}(5,6,7,10,11,15),$	$V_1[X_L^0(0\sim 18, 20\sim 22)\mid X_R^0(0\sim 13, 15\sim 17, 20, 21)\mid X_R^{19}(5)]$	$2^{77.96}$

	$RK^{22}(5\sim 8,10\sim 12,15,16,20)$ $RK^{23}(1,6\sim 13,15\sim 17,20,21)$				
2	$RK_0([9,13],17,21,[4,8]$ $12,16,20,[0\sim 3,$ $5\sim 7,10,11,15])$	$V_2[X_L^1(0\sim 13,15\sim 17,20,21)\mid X_R^1(0\sim 8,10\sim 12,15,16,20)\mid X_R^{19}(5)]$			$2^{83.15}$
3	$RK_1([3,4],8,7,12,11$ $16,15,20,[0\sim 2,5,6,10])$	$V_3[X_L^2(0\sim 8,10\sim 12,15,16,20)\mid X_R^2(0\sim 3,5\sim 7,10,11,15)\mid X_R^{19}(5)]$			$2^{88.62}$
4	$RK_2(3,2,7,6,11,10,15,[0,1,5])$	$V_4[X_L^3(0\sim 3,5\sim 7,10,11,15)\mid X_R^3(0\sim 2,5,6,10)\mid X_R^{19}(5)]$			$2^{90.15}$
5	$RK_3(2,1,6,5,10,0)$	$V_5[X_L^4(0\sim 2,5,6,10)\mid X_R^4(0,1,5)\mid X_R^{19}(5)]$			$2^{90.22}$
6	Other 46 bits master keys	$2^{46}\cdot 2^{36.11}+2^{46-9.44}\cdot 2^{36.11}\approx 2^{82.11}$			$2^{82.11}$
	Number of Guessed key bits: 84 bits Data complexity: 2^{48}	Memory complexity	Vector Counter	$2^{42}\cdot 8$ bits = 2^{42} bytes	Total Time Complexity: $2^{91.6}$
			Key Candidates	$2^{61.67}\cdot 84/8$ bytes$\approx 2^{65.06}$ bytes	

According to the key schedule, the involved round keys $RK_4(0,1,5), RK_5(0), RK_{23}(5)$ can be directly derived from other guessed round key bits.

Table9. Attack procedure on 27-round Simeck64(128)

Step	Guess	Counter			Complexity
1	$RK^{21}(7), RK^{22}(7,8,12),$ $RK^{23}(7\sim 9,12,13,17),$ $RK^{24}(7\sim 10,12\sim 14,17,18,22)$ $RK^{25}(8\sim 15,17\sim 19,22,23,27)$ $RK^{26}(0,9\sim 11,13\sim 20,$ $22\sim 24,27,28)$	$V_1[X_L^0(0\sim 18,20\sim 22,25,26,30)\mid X_R^0(0\sim 13,15\sim 17,20,21,25)\mid X_R^{21}(5)]$			$2^{110.74}$
2	$RK_0(25,20,21,16,17,12$ $13,8,9,4,[0\sim 3,$ $5\sim 7,10,11,15])$	$V_2[X_L^1(0\sim 13,15\sim 17,20,21,25)\mid X_R^1(0\sim 8,10\sim 12,15,16,20)\mid X_R^{21}(5)]$			$2^{100.57}$
3	$RK_1(20,15,16,11,12,7,8$ $3,4,[0\sim 2,5,6,10])$	$V_3[X_L^2(0\sim 8,10\sim 12,15,16,20)\mid X_R^2(0\sim 3,5\sim 7,10,11,15)\mid X_R^{21}(5)]$			$2^{104.96}$
4	$RK_2(3,2,7,6,11,10,15,[0,1,5])$	$V_4[X_L^3(0\sim 3,5\sim 7,10,11,15)\mid X_R^3(0\sim 2,5,6,10)\mid X_R^{19}(5)]$			$2^{106.57}$
5	$RK_3(2,1,6,5,10,0)$	$V_5[X_L^4(0\sim 2,5,6,10)\mid X_R^4(0,1,5)\mid X_R^{19}(5)]$			$2^{106.64}$
6	Other 77 bits master keys	$2^{77}\cdot 2^{35.33}+2^{77-15.67}\cdot 2^{35.33}\approx 2^{112.33}$			$2^{112.33}$
	Number of Guessed key bits: 102 bits Data complexity: 2^{64}	Memory complexity	Vector Counter	$2^{46}\cdot 24$ bits = $2^{47.58}$ bytes	Total Time Complexity: $2^{112.79}$
			Key Candidates	$2^{70.67}\cdot 102/8$ bytes$\approx 2^{74.34}$ bytes	

According to the key schedule, the involved round keys $RK_4(0,1,5), RK_5(0), RK_{25}(7), RK_{26}(7,8,12)$ can be directly derived from other guessed round key bits.

To make it clearer, we summarize our results in the table 10 below.

Table10. Summary of our zero correlation linear cryptanalysis on Simeck

Algorithm	Rounds Attacked	Rounds Details	Data	Memory	Time
Simeck32/64	20	11+5+4	$O(2^{32})$	$2^{39.37}$	$2^{56.65}$
Simeck48/96	24	13+6+5	$O(2^{48})$	$2^{65.06}$	$2^{91.6}$

Simeck64/128	27	15+6+6	$O(2^{64})$	$2^{74.34}$	$2^{112.79}$

6 Conclusion

This paper investigates the security level on Simeck against zero correlation linear cryptanalysis. For Simeck32, currently best result is proposed. During our research, we find that different rotation constants and different key schedules both have effect on the length we can attack with zero correlation linear cryptanalysis. How to choose and even category these parameters is an interesting topic. On the other hand, the security level against other cryptanalytic methods for Simeck is further to be studied.

Acknowledgements

The authors would like to thank the anonymous reviewers for their helpful comments. This work was supported by the National Natural Science Foundation of China under Grant No. 61202491, 61272041, 61272488, 61402523 and 61572516.

References

[1] G. Yang, B. Zhu, V. Suder, M.D. Aagaard, and G. Gong. The Simeck Family of Lightweight Block Ciphers. CHES 2015, to appear (2015).

[2] N. Bagheri. Linear Cryptanalysis of Reduced-Round SIMECK Variants. Cryptology ePrint Archive, Report 2015/716 (2015).

[3] S. Kölbl and A. Roy. A Brief Comparison of Simon and Simeck. Cryptology ePrint Archive, Report 2015/706 (2015).

[4] R. Beaulieu, D. Shors, J. Smith, S. Treatman-Clark, B. Weeks, and L. Wingers. The SIMON and SPECK Families of Lightweight Block Ciphers. Cryptology ePrint Archive, Report 2013/404 (2013).

[5] A. Bogdanov, V. Rijmen. Linear hulls with correlation zero and linear cryptanalysis of block ciphers, Designs, Codes and Cryptography, March 2014, Volume 70, Issue 3, pp. 369-383 (2014).

[6] Q. Wang, Z. Liu, K. Varıcı, Y. Sasaki, V. Rijmen, Y. Todo. Cryptanalysis of Reduced-round SIMON32 and SIMON48. INDOCRYPT 2014. Volume 8885, pp. 143-160 (2014).

[7] J. Alizadeh, H. Alkhzaimi, M.R. Aref, N. Bagheri, P. Gauravaram, A. Kumar, M.M. Lauridsen, S.K. Sanadhya. Cryptanalysis of SIMON Variants with Connections. RFIDSec 2014. Volume 8651, pp. 90-107 (2014).

Appendix A

Table 6. 11-round zero correlation linear distinguisher on Simeck32

Round	Γ_L^r	Γ_R^r
0	1000000000000000	0000000000000000
1	0000000000000000	1000000000000000
2	1000000000000000	?1000?0000000000
3	?1000?0000000000	??100??000?00000
4	??100??000?00000	???10???00??000?
5	???10???00??000?	?????????0???00?

5	???0????????????	??00???????????0?
6	??00???????????0?	?000????10???00?
7	?000????10???00?	00000??100??000?
8	00000??100??000?	00000?1000?00000
9	00000?1000?00000	0000010000000000
10	0000010000000000	0000000000000000
11	0000000000000000	0000010000000000

Table 7. 13-round zero correlation linear distinguisher on Simeck48

Round	Γ_L^r	Γ_R^r
0	10000000000000000000000	00000000000000000000000
1	00000000000000000000000	10000000000000000000000
2	10000000000000000000000	?1000?000000000000000000
3	?1000?000000000000000000	??100??000?0000000000000
4	??100??000?0000000000000	???10???00??000?00000000
5	???10???00??000?00000000	????1????0???00??000?000
6	????1????0???00??000?000	????????????????0???00??00
6	0???0????????????????????	0??00????????????????0???0
7	0??00????????????????0???0	0?000????1????0???00??00
8	0?000????1????0???00??00	00000???10???00??000?000
9	00000???10???00??000?000	00000??100??000?00000000
10	00000??100??000?00000000	00000?1000?0000000000000
11	00000?1000?0000000000000	00000100000000000000000
12	00000100000000000000000	00000000000000000000000
13	00000000000000000000000	00000100000000000000000

Table 8. 15-round zero correlation linear distinguisher on Simeck64

Round	Γ_L^r	Γ_R^r
0	10000000000000000000000000000000	00000000000000000000000000000000
1	00000000000000000000000000000000	10000000000000000000000000000000
2	10000000000000000000000000000000	?1000?000000000000000000000000
3	?1000?000000000000000000000000	??100??000?0000000000000000000
4	??100??000?0000000000000000000	???10???00??000?0000000000000000
5	???10???00??000?0000000000000000	????1????0???00??000?00000000000
6	????1????0???00??000?00000000000	????????????????0???00??000?000000
6	????0????????????????????????????	???00????????????????????????????0
7	???00????????????????????????????0	??000?0????????????????????0???00
8	??000?0????????????????????0???00	?000000?????????????0???00??000
9	?000000?????????????0???00??000	0000000????1????0???00??000?0000
10	0000000????1????0???00??000?0000	0000000???10???00??000?000000000
11	0000000???10???00??000?000000000	0000000??100??000?00000000000000
12	0000000??100??000?00000000000000	0000000?1000?0000000000000000000
13	0000000?1000?0000000000000000000	00000001000000000000000000000000
14	00000001000000000000000000000000	00000000000000000000000000000000
15	00000000000000000000000000000000	00000001000000000000000000000000