Hindawi
Security and Communication Networks
Volume 2022, Article ID 8464960, 15 pages
https://doi.org/10.1155/2022/8464960

Research Article

Related-Key Differential Attacks on Reduced-Round LBlock

Tianling Weng,[1] Tingting Cui ,[1] Ting Yang,[2] and Yinghua Guo[3]

[1]*College of Cyber Security, Hangzhou Dianzi University, Hangzhou 310018, China*
[2]*WuHan Marine Communication Research Institute, Wuhan 430000, China*
[3]*Shandong Big Data Center, Jinan 250000, China*

Correspondence should be addressed to Tingting Cui; cuitingting@hdu.edu.cn

Received 13 February 2022; Revised 5 June 2022; Accepted 4 July 2022; Published 16 September 2022

Academic Editor: Jinguang Han

LBlock, as one of the typical lightweight encryption schemes, is a 32-round block cipher with 64 bit block and 80 bit master key. It can be widely applied in the IoT environment because of its friendly software and hardware implementations. Since it came out, it has encountered many attacks. In this paper, we evaluate LBlock's ability against related-key differential attack more accurately based on SMT method. On the one hand, we propose tighter lower bounds on the minimal number of active S-boxes for up to 19 rounds of LBlock, which are 8 more rounds than previous ones. Then, we propose the upper bounds of total probabilities for up to 19 rounds of LBlock for the first time. On the other hand, with a suitable 17-round related-key differential distinguisher, we propose attacks on 22- and 23-round LBlock. Each of these attacks has lower time complexity and data complexity than previous ones for the same rounds of LBlock.

1. Introduction

In recent years, the Internet of Things (IoT), as the emerging product of Internet, has been widely applied in industry, agriculture, environment and so on, such as smart transportation and smart medical. In IoT, Radio Frequency Identification (RFID), Sensor Network[1], Machine-to-Machine (M2M) and Cloud Computing are key techniques. However, within them, lots of devices have limited resources, which results in the useless of traditional encryption ciphers to guarantee the security of data. To solve this problem, lightweight encryption schemes are more and more popular, such as PRESENT [2], LED [3], HIGHT [4], LBlock [5], SIMON [6], SPECK [6] and SKINNY [7].

As one of the typical lightweight encryption schemes, LBlock is a 32-round block cipher proposed by Wu et al. at ACNS [5]. It adopts a variant Feistel construction with 80 bit master key and 64 bit block. Thanks to its simple round function, it has friendly software and hardware implementations. Its slightly modified version LBlock-s [8] has been used in authenticated encryption scheme LAC in CAESAR competition.

Since LBlock was raised, it has encountered differential attacks [9, 10], integral attacks [11], impossible differential attacks [12] and boomerang attacks [13] in both single-key settings and related-key settings. In single-key settings, the designers of LBlock proposed the first impossible differential attack and integral attack on 20-round LBlock in the design document. Then, the impossible differential attack was further improved in [14, 15], and it could attack 22-round LBlock now.

In related-key settings, the designers first gave valid related-key differential characteristics for up to 13-round LBlock in the design document [5]. At ASIACRYPT'14, Sun et al. [16] used MILP method to evaluate the lower bounds on the minimal active S-boxes of related-key differential characteristics for reduced rounds of LBlock. In the end, they gave security bounds for reduced 1- to 11-round LBlock, although the bounds were a little relax. Later, Sun et al. [17] improved these results and proposed more accurate bounds for the same reduced rounds of LBlock. Besides that, they found a valid characteristic for 15-round LBlock with 23 active S-boxes. Please note that all bounds in these previous works are measured by the number of active S-boxes instead

of the total probability due to high time complexity in the search process, although measuring by probability is more accurate. To recover the key for as many rounds of a cipher as possible, the distinguisher used in the attack usually is expected for as long as possible. How to overcome the shortage in MILP method so as to evaluate more accurate bounds of related-key differential attack on LBlock? This is our first motivation to do this work.

Once a good distinguisher is found, we can go on recovering the secret key out. In [14], Liu et al. proposed the first related-key differential attack on 22-round LBlock based on a 16-round truncated differential. Then, Minier and Naya-Plasencia [18] used related-key impossible differential cryptanalysis to attack 22-round LBlock. Then this work was improved by Wen et al. [19]. They built a new searching algorithm to find out a 16-round related-key impossible differential instead of the 15-round one in [18], so they attacked one more round, i.e. 23 rounds. Please note that although the 23-round attack is suitable for all master keys, it needs 4 different related keys totally. This is a more strict requirement than 2 related keys in traditional related-key (impossible) differential attacks. If we can find longer related-key differential characteristics (more than 16 rounds), when we evaluate more accurate security bounds of LBlock, is there any better related-key differential attack on LBlock? This is our second motivation to do this work.

In this paper, we evaluate the resistance of LBlock against related-key differential attacks by security bounds based on SMT technique [20] and propose better related-key differential attacks on LBlock. Our contributions are as follows:

1.1. Propose More Accurate Bounds of Security against Related-Key Differential Attack on LBlock. In this work, we first propose tighter lower bounds on the minimal number of active S-boxes for up to 19 rounds of LBlock based on the SMT method. Then, we take the total probability of related-key differential characteristic as an objective function to evaluate the ability of LBlock against related-key differential attack and propose the upper bounds of probabilities for up to 19 rounds of LBlock for the first time. All results are summarized in Tables 1–3.

1.2. Propose 22- and 23-Round Related-Key Differential Attacks on LBlock. In this work, we first find out a suitable related-key differential distinguisher for 17-round LBlock with SMT method by a special observation and some constraints on input and output differences. This strategy is similar to that in [21] to some extent, aiming to reduce the search space so as to speed up the search according to some suitable observations. With this distinguisher, we propose attacks on 22- and 23-round LBlock. Each of these attacks has lower time complexity and data complexity than previous ones. All results are summarized in Table 4.

1.2.1. Outline. In Section 2, we briefly recall the specification of LBlock and the description of the SMT method. Then, we propose new security bounds against related-key differential

Table 1: Summary of lower bounds on the minimal number of active S-boxes or upper bound on probability in related-key differential characteristics for reduced-round LBlock (bounds are measured by the number of active S-boxes or total probability).

Objective function	# Rounds	Bounds	Reference
# Active S-boxes	11	10	[16]
# Active S-boxes	13	26	[5]
# Active S-boxes	15	23	[17]
# Active S-boxes	**19**	≤ 31	**Section 3**
Total probability	**19**	$\geq 2^{-75}$	**Section 3**

Table 2: Summary of results about the lower bounds on the number of active S-boxes in related-key differential characteristics under different rounds of LBlock.

Rounds	Lower bounds		
	This paper	[16]	[17]
1	**0**	0	—
2	**0**	0	—
3	**0**	0	—
4	**0**	0	—
5	**1**	1	1
6	**2**	2	2
7	**4**	3	4
8	**6**	5	6
9	**7**	6	8
10	**9**	8	10
11	**11**	10	12
12	**13**	—	—
13	**16**	—	—
14	**18**	—	—
15	**20**	—	23
16	**23**	—	—
17	**26**	—	—
18	**28**	—	—
19	≤ 31	—	—

Table 3: The upper bounds of probabilities for related-key differential characteristics under different rounds of LBlock.

#Rounds	Maximal probability
1	2^{-0}
2	2^{-0}
3	2^{-0}
4	2^{-0}
5	2^{-2}
6	2^{-4}
7	2^{-8}
8	2^{-12}
9	2^{-16}
10	2^{-21}
11	2^{-26}
12	2^{-32}
13	2^{-39}
14	$\geq 2^{-45}$
15	$\geq 2^{-51}$
16	$\geq 2^{-55}$
17	$\geq 2^{-63}$
18	$\geq 2^{-70}$
19	$\geq 2^{-75}$

Table 4: Summary of attacks on LBlock.

Attack type	#Rounds	Time	Data	#Keys per attack	#Weak keys	Reference
Boomerang	18	$2^{70.84}$	$2^{63.27}$	1	2^{80}	[22]
ID	20	$2^{72.7}$	2^{63}	1	2^{80}	[5]
Integral	20	$2^{63.7}$	$2^{63.7}$	1	2^{80}	[5]
ID	21	$2^{73.7}$	$2^{62.5}$	1	2^{80}	[14]
ID	21	$2^{69.5}$	2^{63}	1	2^{80}	[15]
ID	22	$2^{79.28}$	2^{58}	1	2^{80}	[15]
ID	23	$2^{71.8}$	2^{59}	1	2^{80}	[23]
RKD	22	2^{67}	$2^{63.1}$	2	2^{78}	[24]
RKID	22	2^{70}	2^{47}	4	2^{78}	[18]
RKD	**22**	$\mathbf{2^{45.54}}$	$\mathbf{2^{61}}$	**2**	$\mathbf{2^{70}}$	**Section 4**
RKID	23	$2^{78.3}$	$2^{61.4}$	4	2^{80}	[19]
RKD	**23**	$\mathbf{2^{65.48}}$	$\mathbf{2^{61}}$	**2**	$\mathbf{2^{70}}$	**Section 4**

Remark: ID denotes impossible differential; RKD denotes related-key differential; RKID denotes Related-key impossible differential.

attack on LBlock in Section 3. In Section 4, we find out a suitable 17-round related-key differential distinguisher and use it to attack 22- and 23-round LBlock. In the end, we conclude the paper in Section 5.

2. Preliminaries

In this part, we first introduce some notations used in this paper, then briefly recall the specification of LBlock and the description of Simple theorem Prover (STP) Solver.

2.1. Notations. In this paper, some notations are defined as follows:

Notation	Definition
ΔX_i	32 bits of 64 bit input difference
$X_i \Vert X_{i-1}$	64 bit input of round i
Y_i	32 bit input of S-box layer in round i
Z_i	32 bit output of S-box layer in round i
K	80 bit master key
K_i	32 bit subkey used in round i
$k_{i\sim j}$	The i-th to j-th bits of master key, $0 \le i, j \le 79$
k_i^j	The j-th nibble of subkey K_i, $j = 0, 1, \ldots, 7$
F	Round function of LBlock
$\oplus$	Exclusive-OR
$\lll i$	Left rotate i bits
$a \Vert b$	Concatenation of two binary strings a, b
$[i]_2$	Binary form of an integer i

2.2. Specification of LBlock. LBlock [5] is a 32-round lightweight block cipher proposed by Wu et al. at ACNS 2011. It adopts a variant Feistel construction with 80 bit master key and 64 bit block. LBlock's round function is shown in Figure 1.

As we can see from Figure 1, the 64 bit input of the i-th round function is divided into two branches named as X_i and X_{i-1}, then is updated with the following equation to generate 64 bit output $X_{i+1} \Vert X_i$, $i = 1, 2, \ldots, 32$:

$$X_{i+1} = F(X_i) \oplus (X_{i-1} \lll 8). \quad (1)$$

Here symbol F function consists of three parts: AddRoundKey (AK), nonlinear S-box layer (S) and nibble permutation layer (P), which are listed as follows:

(i) AddRoundKey (AK): The secret subkey K_i is mixed with input X_i by XOR operation.

(ii) S-box layer (S): This layer includes 8 different parallel 4×4 S-boxes $S_0 \sim S_7$ whose specifications are shown in Table 5.

(iii) Nibble Permutation layer (P): This layer is a linear symmetric nibble-wise permutation.

Please note that there is no branch-wise permutation in the last round function and LBlock has totally 32 rounds, so we can denote plaintext as $X_1 \Vert X_0$ and ciphertext as $X_{32} \Vert X_{33}$.

2.2.1. Key Schedule Algorithm. The master key K of LBlock has 80 bits, which is stored in a register at very beginning. Without loss of generality, we also denote the master key as $K = k_{79}k_{78} \ldots k_1k_0$. The leftmost 32 bits of master key are used as the first subkey K_0, then the current register is updated with the following operations in Algorithm 1.

Here S_8 and S_9 are two 4 bit S-boxes, which are shown in Table 5. For more details about LBlock, please refer to [5].

2.3. Simple theorem Prover. During the last few years, automatic search tools are more and more widely used to search for differential characteristics or linear approximations. One of the tools is Simple theorem Prover (STP) which bases on Satisfiability Modulo Theories (SMT) method. STP [20] is an effective SMT solver originally designed to solve the constraints of bit-vectors and arrays. When searching for differential characteristics with the STP solver, users usually transform such problems into a series of constraints. This solver has been used for cryptanalysis, such as [25–28]. CVC language [20] is STP's default language. Here, we use Table 6 to list some orders in the CVC language. For more details, please refer to https://stp.readthedocs.io/en/latest/cvc-input-language.html.

In order to describe the STP model clearly, we also give two simple examples as follows.

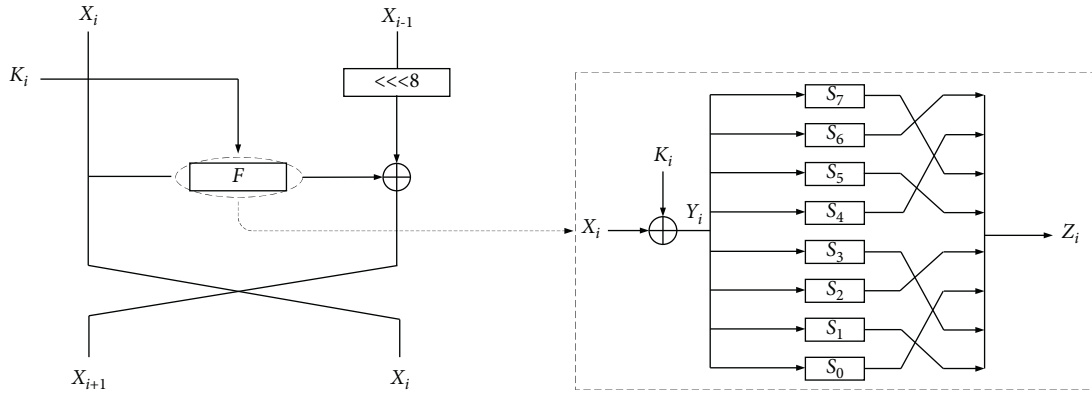

Figure 1: Round function of LBlock.

Table 5: S-boxes used in LBlock.

S_0	14, 9, 15, 0, 13, 4, 10, 11, 1, 2, 8, 3, 7, 6, 12, 5
S_1	4, 11, 14, 9, 15, 13, 0, 10, 7, 12, 5, 6, 2, 8, 1, 3
S_2	1, 14, 7, 12, 15, 13, 0, 6, 11, 5, 9, 3, 2, 4, 8, 10
S_3	7, 6, 8, 11, 0, 15, 3, 14, 9, 10, 12, 13, 5, 2, 4, 1
S_4	14, 5, 15, 0, 7, 2, 12, 13, 1, 8, 4, 9, 11, 10, 6, 3
S_5	2, 13, 11, 12, 15, 14, 0, 9, 7, 10, 6, 3, 1, 8, 4, 5
S_6	11, 9, 4, 14, 0, 15, 10, 13, 6, 12, 5, 7, 3, 8, 1, 2
S_7	13, 10, 15, 0, 14, 4, 9, 11, 2, 1, 8, 3, 7, 5, 12, 6
S_8	8, 7, 14, 5, 15, 13, 0, 6, 11, 12, 9, 10, 2, 4, 1, 3
S_9	11, 5, 15, 0, 7, 2, 9, 13, 4, 8, 1, 12, 14, 10, 3, 6

Example 1. Describe $a = b \oplus c$:

a: BITVECTOR(4);

b: BITVECTOR(4);

c: BITVECTOR(4);

ASSERT(a = BVXOR(b, c));

The first three statements are to claim that a, b, c are 4 bit variables.

Example 2. S_0 is an S-box used in LBlock, the Differential Distribution Table (DDT) of S_0 can be described in CVC language as:

S_0: ARRAY BITVECTOR(8) OF BITVECTOR(1);

ASSERT(S_0[0bin00000000] = 0bin1);

ASSERT(S_0[0bin00000000] = 0bin0);

⋮

ASSERT(S_0[0bin11111111] = 0bin0);

In this Example 2, BTVECTOR(i) denotes that the variable has i bits. ARRAY BITVECTOR (i) OF BITVECTOR(j) denotes that the input of the array is i bits, and the output is j bits. In the statement of ASSERT(S_0[0bin$x_1x_2 \ldots x_8$] = 0bin$y$), the 8 bit $x_1x_2 \ldots x_8$ includes 4 bit input difference $x_1x_2x_3x_4$ and 4 bit output difference $x_5x_6x_7x_8$. If this differential pattern $x_1x_2x_3x_4 \longrightarrow x_5x_6x_7x_8$ is possible in DDT. Then $y = 1$, otherwise $y = 0$.

3. Evaluate the Resistance of LBlock against Related-Key Differential Attack by Bounds

Taking the total number of active S-boxes as the objective function is a very traditional method to evaluate the resistance of such cipher against related-key differential attack by designers because it is more easier and less time-consuming, especially for byte-wise ciphers. However, taking the maximal probability as the objective function directly results in more precise evaluation. Unluckily, it is more complex and time-consuming, especially for high rounds of a cipher. Attackers usually would like this method to find the best related-key differential characteristic so as to carry out a good attack. In this work, we evaluate the security of LBlock with both ideas.

In this section, we use the SMT method to automatically evaluate the resistance of LBlock against related-key differential attack by computing the exact lower bounds on the minimal number of active S-boxes and the upper bounds of the maximal probabilities. As a result, we find out the minimal number of active S-boxes for $1 \sim 18$ rounds and a lower bound on the number of active S-boxes for 19-round related-key differential characteristics in Subsection 3.1, as well as the maximal probabilities for $1 \sim 13$ rounds and upper bounds of probabilities for $14 \sim 19$ rounds of related-key differential characteristics in Subsection 3.2.

3.1. Lower Bounds on the Minimal Number of Active S-Boxes. Computing the lower bounds on the minimal number of active S-boxes in related-key differential characteristics under different rounds of target cipher, is a common and traditional method to evaluate the ability of a cipher against related-key differential attack. In [16], Sun et al. first proposed bounds on the number of active S-boxes in related-key differential characteristics under different rounds of LBlock. However, because of the time complexity, they could only evaluate under at most 11 rounds of LBlock. Besides that, they found the differential characteristics with the lower bounds on the number of active S-boxes instead of maximal probability for block cipher with 4×4 S-boxes. Later, these results were improved in [17] based on MILP method. They gave the exact lower bounds on the number of active S-boxes

(1) $K_0 = k_{79} \ldots k_{48}$
(2) **for** i in $1, 2, \ldots, 31$ **do**
$K \lll 29$
$[k_{79}k_{78}k_{77}k_{76}] \leftarrow S_9[k_{79}k_{78}k_{77}k_{76}]$
$[k_{75}k_{74}k_{73}k_{72}] \leftarrow S_8[k_{75}k_{74}k_{73}k_{72}]$
$[k_{50}k_{49}k_{48}k_{47}k_{46}] \leftarrow [k_{50}k_{49}k_{48}k_{47}k_{46}] \oplus i$
$K_i \leftarrow k_{79}k_{78} \ldots k_{48}$
end

Algorithm 1: Key schedule algorithm.

Table 6: Description of some common orders in CVC language.

Name	Symbol	Example
Concatenation	@	$t_1@t_2 \ldots @t_m$
Extraction	$[i: j]$	$x[31: 26]$
Left shift	$\ll$	0bin0011 $\ll$ 3 = 0bin0011000
Bitwise AND	&	$t_1 \& t_2 \& \ldots \& t_m$
Bitwise XOR	BVXOR	BVXOR(t_1, t_2)
Bitvector add	BVPLUS	BVPLUS$(n, t_1, t_2, \ldots, t_m)$
Greater than or equal to	BVGE	BVGE(t_1, t_2)

for $5 \sim 11$ rounds of LBlock. Similarly, because of high time complexity, they only could try to find the bound for high rounds and gave a bound for 15 rounds of LBlock (the program did not run over). Since the MILP-based method which they used could not find the differential characteristic with the maximal probability in practical time, the bounds they proposed still were not the exact lower bounds. All these results are summarized in Table 2.

In this work, we use the SMT method to automatically search for the exact lower bounds on the number of active S-boxes in related-key differential characteristics for $1 \sim 19$ rounds of LBlock. Firstly, we describe the differential propagations on encryption scheme and key schedule algorithm of LBlock by an STP model in CVC language and add the objective function as a constraint to limit the number of active S-boxes in the model. The whole process is summarized in Algorithm 2. By solving the STP model, the exact lower bounds on the number of active S-boxes for $1 \sim 19$ rounds of LBlock can be solved out, which are shown in Table 2.

As we can see from Table 2, our results are more accurate and strong to show the actual ability of LBlock against related-key differential attack compared with previous related works. Firstly, our work shows there is valid related-key differential characteristic for 19-round LBlock for the first time. As an example, we list one related-key differential characteristic for 19-round LBlock in Table 7. Secondly, we find that the exact lower bound of n-round related-key differential characteristic has 2 or 3 active S-boxes more than the exact lower bound of $n-1$-round related-key differential characteristic, where $7 \leq n \leq 18$. In other words, there may be at least 40 active S-boxes for 23-round LBlock, which means there is no valid related-key differential characteristic for 23-round LBlock.

Table 7: 19-round related-key differential characteristic with 31 active S-boxes (#AS means the number of active S-boxes.).

Rd_i	ΔX_i	ΔX_{i-1}	ΔK_i	#AS
1	30000004	400010E3	00003000	3
2	0000E800	30000004	00003800	1
3	00000400	0000E800	00000400	0
4	00E80000	00000400	00180000	1
5	000A0000	00E80000	001C0000	2
6	E0090000	000A0000	00020000	3
7	0C800000	E0090000	0C000000	2
8	090E00E0	0C800000	06000000	4
9	06000000	090E00E0	06000030	1
10	0E00E008	06000000	00000018	3
11	500000A4	0E00E008	00000018	3
12	0000000C	500000A4	00001800	3
13	00000C00	0000000C	00000C00	0
14	00000C00	00000C00	00000C00	0
15	000C0000	00000C00	000C0000	0
16	000C0000	000C0000	00060000	1
17	00000000	000C0000	00060000	2
18	0F000000	00000000	0F000000	1
19	00000000	0F000000	09000000	1
20	3000000F	00000000	—	—

3.2. Upper Bounds of the Maximal Probabilities. In Subsection 3.1, we give the exact lower bounds on the minimal number of active S-boxes for $1 \sim 19$ rounds of LBlock by the SMT method to evaluate the ability of LBlock against the related-key differential attack. However, although we have obtained the minimal number of active S-boxes under different rounds of LBlock, we can not directly determine the maximal probability under different rounds. For example, the minimal number of active S-boxes of 12-round related-key differential characteristic is 13. Among such characteristics, their probabilities are in range from 2^{-26} to 2^{-39}, but actually, the maximal probability for 12-round LBlock is 2^{-32} (totally 13 active S-boxes). In other words, evaluating the ability against related-key differential attack with the maximal probability can get tighter security bounds, compared with evaluating with the minimal number of active S-boxes. Unfortunately, there is no previous work on searching the upper bound of probabilities for related-key differential characteristics under different rounds of LBlock. In this work, we evaluate the upper bounds of probabilities under different rounds of LBlock by the SMT method.

Table 8: Related-key differential characteristic for 13-round LBlock (total probability is 2^{-39}).

Rd_i	ΔX_i	ΔX_{i-1}	ΔK_i	pro_i
1	00000200	00000020	00000000	2^{-2}
2	00000000	00000200	00000000	2^{-0}
3	00020000	00000000	00120000	2^{-3}
4	00070000	00020000	00000000	2^{-2}
5	00000000	00070000	00000000	2^{-3}
6	07000000	00000000	07000000	2^{-0}
7	00000000	07000000	00000000	2^{-0}
8	00000007	00000000	0000001C	2^{-8}
9	0000010F	00000007	D0000000	2^{-8}
10	00A06000	0000010F	00000000	2^{-4}
11	00000F40	00A06000	00000F40	2^{-0}
12	A0600000	00000F40	00000000	2^{-5}
13	00204000	A0600000	00000000	2^{-4}
14	60010060	00204000	—	—

In fact, taking the upper bound of probability as the objective function instead of the number of active S-boxes will slow down the searching speed, especially for high rounds of LBlock.

Luckily, by observing the best related-key differential characteristics for short rounds of LBlock, we find an observation as follows.

Transforming this observation as a constraint into the STP model can speed up the STP solver and get tighter results for 14 ~ 19 rounds.

Observation 1 Assume $R_1 \longrightarrow R_2 \longrightarrow \cdots \longrightarrow R_r\,(7 \le r \le 13)$ is an r-round related-key differential characteristic with the maximal probability and $P_i\,(1 \le i \le r-1)$ is the probability on the i-th round, there are at least two consecutive rounds with probability of 1. i.e. $P_i = 1 \text{and} P_{i+1} = 1$.

For example, we list one of the best related-key differential characteristic for 13-round LBlock in Table 8. From this Table, we can see that the probability is 1 on both round 6 and 7.

We transform the above observation in CVC language as a constraint and add it to the STP model. After that, we can apply Algorithm 3 (similar to Algorithm 2) to search the upper bounds of probabilities for related-key differential characteristics under different rounds of LBlock. The results are shown in Table 3.

Please note that the probability in Table 3 denotes the total probability including both probabilities on cipher itself and on the key schedule. From Table 3, we can see that the best probability of related-key differential charateristic for 19-round LBlock is no less than 2^{-75}.

As an example, we give one related-key differential characteristic with high probablity for 19-round LBlock in Table 9.

Table 9: Related-key differential characteristic for 19-round LBlock (total probability is 2^{-75}).

Rd_i	ΔX_i	ΔX_{i-1}	ΔK_i	pro_i
1	B0000004	C000209E	00003000	2^{-6}
2	00009800	B0000004	00003800	2^{-2}
3	00000400	00009800	00000400	2^{-0}
4	00980000	00000400	00180000	2^{-3}
5	000A0000	00980000	001C0000	2^{-4}
6	90090000	000A0000	00020000	2^{-7}
7	0C800000	90090000	0C000000	2^{-6}
8	090E0090	0C800000	06000000	2^{-10}
9	06000000	090E0090	06000030	2^{-2}
10	0E009008	06000000	00000018	2^{-7}
11	500000A4	0E009008	00000018	2^{-7}
12	0000000C	500000A4	00001800	2^{-9}
13	00000C00	0000000C	00000C00	2^{-0}
14	00000C00	00000C00	00000C00	2^{-0}
15	000C0000	00000C00	000C0000	2^{-0}
16	000C0000	000C0000	00060000	2^{-3}
17	00000000	000C0000	00060000	2^{-5}
18	0E000000	00000000	0E000000	2^{-2}
19	00000000	0E000000	05000000	2^{-2}
20	4000000E	00000000	—	—

4. Related-Key Differential Attacks on LBlock

In this section, we propose the best related-key differential attacks on LBlock as far as we know. Firstly, we search out a suitable 17-round related-key differential distinguisher on LBlock by combining the techniques used in Section 3.2 and some extra constraints, which are described in Subsection 4.1. Then, based on this distinguisher, we propose the best related-key differential attacks on 22-round and 23-round LBlock in Subsections 4.2 and 4.3 respectively.

4.1. Automatic Search for Suitable Related-Key Differential Distinguishers on LBlock. A related-key differential attack consists of a suitable related-key differential characteristic and a key-recovery phase. In most cases, the longer rounds an attack achieves on a cipher, the better such attack is. Thus, the related-key differential distinguisher is expected as long as possible, while the corresponding key-recovery phase is expected to involve more rounds. However, there is a tradeoff between the related-key differential distinguisher and key-recovery phase. The longest related-key differential characteristic with the largest probability may not result in the best attack because it may lead to fast diffusion and confusion in the key-recovery phase, further leading to attack fewer rounds. Generally, In order to enhance the effect in the key-recovery phase, few active nibbles are expected on input and output differences of related-key differential distinguisher.

In this part, we combine the automatic search method used in section 3.2 and extra constraints on input and output differences of distinguisher to search out a good related-key differential distinguisher which is very suitable to implement the best attack on LBlock. In detail, some requirements and constraints in the automatic search method of suitable r-round related-key differential distinguisher are listed as follows:

(i) Assume the input and output differences of target r-round related-key differential characteristic are $(\Delta X_1, \Delta X_0)$ and $(\Delta X_{r+1}, \Delta X_r)$ respectively, we require that $Hw_{\text{nibble}}(\Delta X_i) \le 2, i = 0, 1, r, r+1$, where

```
(1) Function Encryption Algorithm:
(2)   r: the number of rounds
(3)   eqn = [ ] //list
(4)   for i ← 0 to r − 1 do
(5)     Add "ASSERT(ΔY_i = BVXOR(ΔX_i, ΔK_i));" in eqn
(6)     Add "ASSERT(S7[ΔY_i[31: 28]@ΔZ_i[23: 20]] = 0bin1);" in eqn
(7)      ⋮
(8)     Add "ASSERT(S0[ΔY_i[3: 0]@ΔZ_i[11: 8]] = 0bin1);" in eqn
(9)     Add "ASSERT(ΔX_{i−1}[23: 0]@ΔX_{i−1}[31: 24] = BVXOR(ΔZ_i, ΔX_{i+1}));" in eqn
        //Note: These variables are from Figure 1
(10)  end
(11)  return eqn
(12)
(13) Function Key Schedule Algorithm:
(14)  r: the number of rounds
(15)  eqn = [ ] //list
(16)  for i ← 0 to r − 1 do
(17)    Add "ASSERT((S8[ΔK_i[50: 0]@ΔK_i[79: 51]])[75: 72]@ΔK_{i+1}[75: 72] = 0bin1);" in eqn
(18)    Add "ASSERT((S9[ΔK_i[50: 0]@ΔK_i[79: 51]])[79: 76]@ΔK_{i+1}[79: 76] = 0bin1);" in eqn
(19)    Add "ASSERT((ΔK_i[50: 0]@ΔK_i[79: 51]))[71: 70] = ΔK_{i+1}[71: 0]); " in eqn
(20)  end
(21)  return eqn
(22)
(23) Function Main:
(24)  Create a file LBlock.cvc
(25)  eqn = [ ] //list
(26)  Add "S0: ARRAY BITVECTOR(8) OF BITVECTOR(1);
(27)   ⋮
(28)  S9: ARRAY BITVECTOR(8) OF BITVECTOR(1);
(29)  ASSERT(S0[0bin00000000] = 0bin1);
(30)  ASSERT(S0[0bin00000001] = 0bin0);
(31)   ⋮
(32)  ASSERT(S9[0bin11111110] = 0bin1);
(33)  ASSERT(S9[0bin11111111] = 0bin0);" in eqn
      //Describe the differential distribution table of S_0 ~ S_9

(34)  eqn + = Encryption Algorithm();
(35)  eqn + = Key Schedule Algorithm();
(36)
(37)  Add "ASSERT(BVGT(K_1, 0bin0 . . . 0)); " in eqn

(38)  Add " ASSERT(BVGT(BVPLUS(8, 0bin0000000@((k_1[50: 0]@k_1[79: 51])[75: 72][3: 3]
      |(k_1[50: 0]@k_1[79: 51])[75: 72][2: 2]|(k_1[50: 0]@k_1[79: 51])[75: 72][1: 1]|(k_1[50: ]]@k_1[79: 51])
      [75: 72][0: 0]), . . . , 0bin0000000@
      (X_{r−1}[3: 0][3: 3]|X_{r−1}[3: 0][2: 2]|X_{r−1}[3: 0][1: 1]|X_{r−1}[3: 0][0: 0])), 0bin00000000)
      ;" in eqn

(39)  write eqn in LBlock.cvc
(40)
(41) stp LBlock.cvc
```

Algorithm 2: Get the exact lower bound on the number of active S-boxes by STP model.

$Hw_{\text{nibble}}(X)$ denotes the number of active nibbles in state X;

(ii) There exist consecutive two rounds with probability 1 in the related-key differential characteristic, which is exactly the requirement in Observation 1;

(iii) In order to adapt key-recovery attack on most cases of masterkey, we limit the probability on the key schedule is $\text{pro}_{\text{key}} \geq 2^{-10}$;

(iv) General requirement: the number of active nibbles on masterkey ΔK satisfies $Hw_{\text{nibble}}(\Delta K) \geq 1$;

```
(1) Function Encryption Algorithm:
(2)   r: the number of rounds
(3)   eqn = [ ] //list
(4)   for i ← 0 to r − 1 do
(5)     Add "ASSERT(ΔY_i = BVXOR(ΔX_i, ΔK_i));" in eqn
(6)     Add "ASSERT(NOT(S7[ΔY_i[31: 28]@ΔZ_i[23: 20]] = 0bin01));" in eqn
(7)      ⋮
(8)     Add "ASSERT(NOT(S0[ΔY_i[3: 0]@ΔZ_i[11: 8]] = 0bin01));" in eqn
(9)     Add "ASSERT(ΔX_{i−1}[23: 0]@ΔX_{i−1}[31: 24] = BVXOR(ΔZ_i, ΔX_{i+1})); " in eqn
(10)  end
(11)  return eqn
(12)
(13) Function Key Schedule Algorithm:
(14)  r: the number of rounds
(15)  eqn = [ ] //list
(16)  for i ← 0 to r − 1 do
(17)    Add "ASSERT(NOT((S8[ΔK_i[50: 0]@ΔK_i[79: 51]])[75: 72]@ΔK_{i+1}[75: 72] = 0bin01));" in eqn
(18)    Add "ASSERT(NOT((S9[ΔK_i[50: 0]@ΔK_i[79: 51]])[79: 76]@ΔK_{i+1}[79: 76] = 0bin01));" in eqn
(19)    Add "ASSERT((ΔK_i[50: 0]@ΔK_i[79: 51]))[71: 0] = ΔK_{i+1}[71: 0]);" in eqn
(20)  end
(21)  return eqn
(22)
(23) Function Main:
(24)  Create a file LBlock.cvc
(25)  eqn = [ ] //list
(26)  Add "S0: ARRAY BITVECTOR(8) OF BITVECTOR(2);
(27)   ⋮
(28)  S9: ARRAY BITVECTOR(8) OF BITVECTOR(2);
(29)  ASSERT(S0[0bin00000000] = 0bin00);
(30)  ASSERT(S0[0bin00000000] = 0bin01);
(31)   ⋮
(32)  ASSERT(S9[0bin00000000] = 0bin11);
(33)  ASSERT(S9[0bin11111111] = 0bin01);" in eqn
(34)  eqn += Encryption Algorithm();
(35)  eqn += Key Schedule Algorithm();
(36)
(37)  Add "ASSERT((Y_0 = 0x00000000 AND Y_1 = 0x00000000) OR ... OR (Y_{r−1} = 0x00000000 AND Y_r = 0x00000000));" in eqn

(38)  Add "ASSERT(BVGT(K_1, 0bin0 ··· 0));" in eqn

(39)  Add "ASSERT(BVGT)(BVPLUS)(10, 0bin00000000@)(S8[(k1[50: 0]@k1[79: 51])[75: 72]@k2[75: 72]),
      0bin00000000@(S9[(k1[50: 0]@k1[79: 51])[79: 76])
      @k2[79: 76]]), ..., 0bin00000000@(S1[Y_r[7: 4]@Z_r[3: 0]]), 0bin00000000
      @(S0[Y_r[3: 0]@Z_r[11: 8]])), 0bin00000000)" in eqn

(40)  write eqn in LBlock.cvc 19
(41)
(42) stp LBlock.cvc
```

Algorithm 3: Get the upper bounds of probability by STP model.

(v) Objective function: maximize the probability on cipher itself $\text{pro}_{\text{cipher}}$.

As a result, we finally find a suitable 17-round related-key differential characteristic, whose total probability is 2^{-70} including 2^{-60} on cipher itself and 2^{-10} on the key schedule. This characteristic is listed in Table 10.

In this 17-round related-key differential characteristic, there are 4 active S-boxes within the key schedule that are distributed on the S_8 of rounds 6, 7, 8 and 17. In round 6, the differential pattern passing S_8 is $1100 \longrightarrow^{S_8} 1100$ with probability 2^{-3}. According to the difference distribution table of S_8, there are two cases for $(k_6^1, k_6'^1)$ that is

Table 10: 17-round related-key differential characteristic with total probability pro = 2^{-70} including $\text{pro}_{\text{cipher}} = 2^{-60}$ on cipher itself and $\text{pro}_{\text{key}} = 2^{-10}$ on the key schedule.

Rd_i	ΔX_i	ΔX_{i-1}	ΔK_i	pro_i
1	00000800	00100003	00000800	2^{-0}
2	10000300	00000800	00000800	2^{-5}
3	00782000	10000300	00180000	2^{-4}
4	00070000	00782000	00040000	2^{-2}
5	70200000	00070000	00040000	2^{-9}
6	0D410000	70200000	0C000000	2^{-8}
7	01060070	0D410000	06000000	2^{-9}
8	06000000	01060070	06000030	2^{-2}
9	06007008	06000000	00000018	2^{-6}
10	100000E4	06007008	00000018	2^{-9}
11	0000000C	100000E4	00001800	2^{-8}
12	00000C00	0000000C	00000C00	2^{-0}
13	00000C00	00000C00	00000C00	2^{-0}
14	000C0000	00000C00	000C0000	2^{-0}
15	000C0000	000C0000	00060000	2^{-3}
16	00000000	000C0000	00060000	2^{-5}
17	0E000000	00000000	0E000000	2^{-0}
18	00000000	0E000000	-	-

Table 11: Differential patterns on active S-boxes in the key schedule of 17-round related-key differential characteristic.

Rd_i	Δ_{in}	Δ_{out}	pro	Input pairs $(k_i^1, k_i'^1)$
6	1100	1100	2^{-3}	$(0111, 1011)$, $(1011, 0111)$
7	0010	0110	2^{-2}	$(0000, 0010)$, $(0010, 0000)$, $(1100, 1010)$, $(1010, 1100)$
8	0010	0110	2^{-2}	$(0000, 0010)$, $(0010, 0000)$, $(1100, 1010)$, $(1010, 1100)$
17	0110	1110	2^{-3}	$(1011, 1101)$, $(1101, 1011)$

$(0111, 1011)$ and $(1011, 0111)$, where k_i^j means the j-th nibble of subkey in round i. Similarly, we can analyze the differential patterns on S_8 in rounds 7, 8 and 17, which are summarized in Table 11.

4.2. Related-Key Differential Attack on 22-Round LBlock. In subsection 4.1, we search out an expected 17-round related-key differential distinguisher on LBlock, whose total probability is 2^{-70} including 2^{-60} on cipher itself and 2^{-10} on the key schedule. In this subsection, we extend three rounds before this distinguisher and two rounds behind this distinguisher to attack 22-round LBlock. It is shown in Figure 2.

4.2.1. Data Collection. From Figure 2, we can see that the forms of differences on plaintext and ciphertext are $\triangle P = (0 * 00 * 0 * *, * * * 0000 *)$ and $\triangle C = (* 000000E, * 0 * 00 * 0 *)$ respectively, where $*$ denotes any 4 bit value. According to the form of difference on plaintext, we can build 2^x structures and each structure involves 2^{32} plaintexts. Totally, the data complexity is 2^{x+32} chosen plaintexts.

Since each structure composes 2^{64} plaintext pairs, there are expected $2^{x+64-32-60} = 2^{x-28}$ plaintext pairs to satisfy the 17-round distinguisher on both input and output differences.

4.2.2. Filtering on Ciphertext. According to the form of ciphertext difference $\triangle C$, there are on average $2^{x+64-44} = 2^{x+20}$ pairs remained after the filtering on ciphertext.

4.2.3. Filtering on Rounds $1 \sim 3$. In this part, we guess the subkey nibbles k_1^1, k_1^0, k_1^6, k_2^7, k_1^3, k_2^1, k_2^0, k_2^2, k_3^3, k_3^0 and k_3^5 one by one to filter pairs. Which are summarized within steps $1 \sim 11$ in Table 12. Taking step 1 as an example, we explain how we filter pairs in detail.

In this step, we need to guess 4 bits subkey k_1^1 corresponding to the master key $k_{55\sim52}$. Note that we know $k_1^1 = k_{55\sim52}$ according to the key schedule algorithm of LBlock. Since k_{54}, k_{53} and k_{52} are already known, it is only necessary to guess k_{55} so as to obtain the value of k_1^1.

Please note that $k_{54\sim52}$ has 2 possible cases, which is equivalent to guessing one more bit. Under each out of 2^2 possible values of k_1^1, we can calculate the output difference of S_7 in round 1 for each plaintext pair, and check whether it equals the third nibble of ΔX_0. If not, we filter out the current plaintext pair. Otherwise, we remain it and go next step. Totally, it needs $2 \times 2^2 \times 2^{x+20}$ operations on Sbox in step 1, while 2^{x+16} plaintext pairs are remained on average after step 1 because the probability is 2^{-4}. The similar idea is applied on steps $2 \sim 11$.

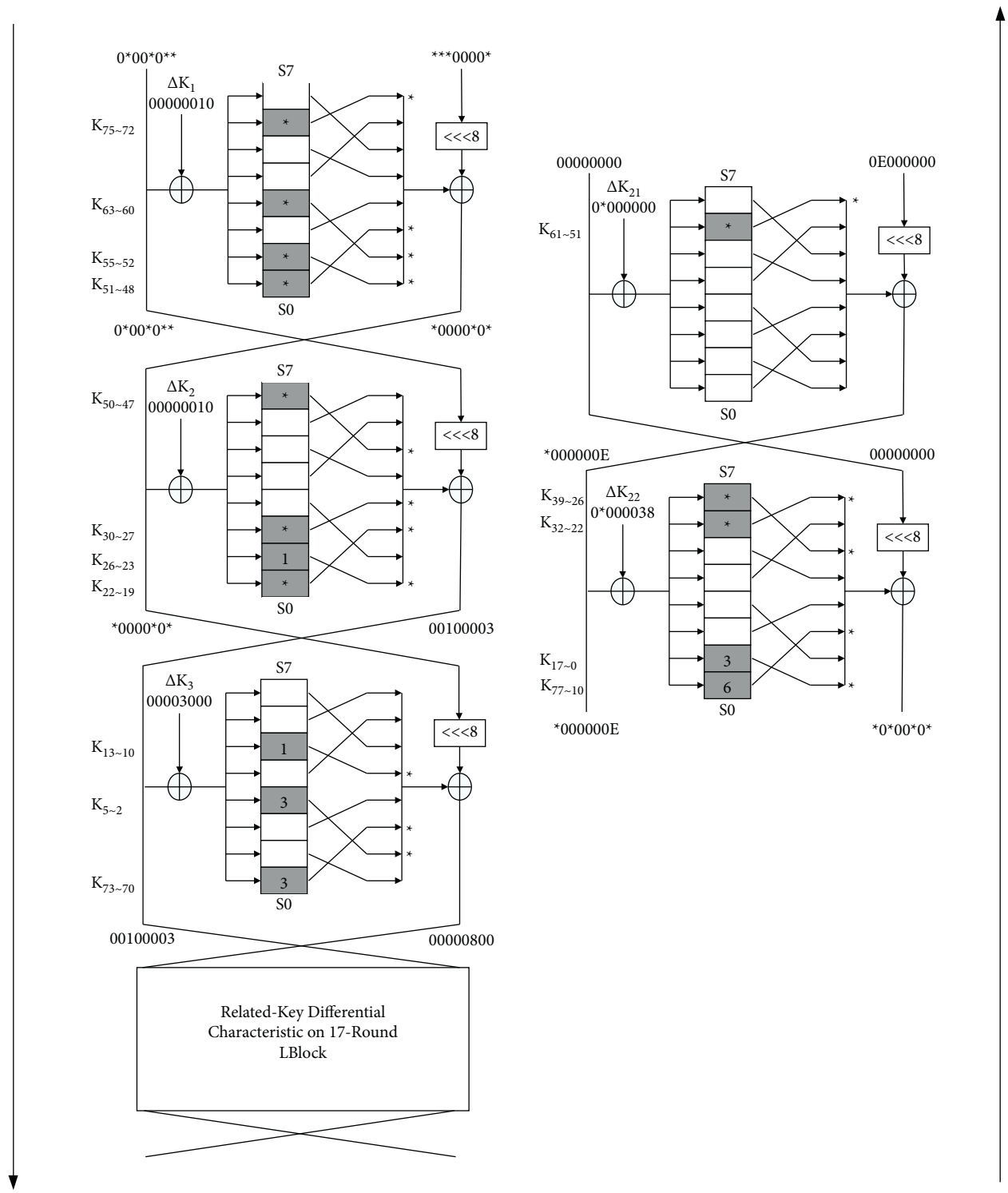

Figure 2: Related-key differential attack on 22-round LBlock.

Sometimes, instead of considering the subkey nibbles independently, we would like to focus on their related masterkey bits according to the key schedule. Thanks to the slow diffusion of LBlock's key schedule, the corresponding masterkey bits of $k_1^1, k_1^0, k_1^6, k_2^7, k_1^3, k_2^1, k_2^0, k_2^2, k_3^3, k_3^0$ and k_3^5 are $k_{55\sim52}, k_{51\sim48}, k_{75\sim72}, k_{50\sim47}, k_{63\sim60}, k_{26\sim23}, k_{22\sim19}, k_{30\sim27}, k_{78\sim1}$, $k_{73\sim70}$ and $k_{13\sim10}$, respectively. However, please note that there are only 2 cases for $k_{3\sim0}$, 4 cases for $k_{54\sim51}$ and 4 cases for $k_{25\sim22}$ because of the 17-round related-key differential distinguisher. Thus, in step 1, we only need to guess k_{55} and 2 cases of $k_{54\sim51}$ (We all know that there are two encryption machines with different masterkeys in related-key setting.

Table 12: Related-key differential attack process on 22-round LBlock.

Step	Subkey	Masterkey	#Bits	Time	#Pairs left
1	k_1^1	$k_{55\sim52}$	2 (k_{55}, 2 cases for $k_{54\sim51}$)	$2\times2^2\times2^{x+20}$	2^{x+16}
2	k_1^0	$k_{51\sim48}$	3 $k_{50\sim48}$	$2\times2^5\times2^{x+16}$	2^{x+12}
3	k_1^6	$k_{75\sim72}$	4	$2\times2^9\times2^{x+12}$	2^{x+8}
4	k_2^7	$k_{50\sim47}$	1 (k_{47})	$2\times2^{10}\times2^{x+8}$	2^{x+4}
5	k_1^3	$k_{63\sim60}$	4	$2\times2^{14}\times2^{x+4}$	2^x
6	k_2^1	$k_{26\sim23}$	3 (k_{26}, 4 cases for $k_{25\sim22}$)	$2\times2^{17}\times2^x$	2^{x-4}
7	k_2^0	$k_{22\sim19}$	3 ($k_{21\sim19}$)	$2\times2^{20}\times2^{x-4}$	2^{x-8}
8	k_2^2	$k_{30\sim27}$	4	$2\times2^{24}\times2^{x-8}$	2^{x-12}
9	k_3^3	$k_{5\sim2}$	3 ($k_{5\sim4}$, 2 cases for $k_{3\sim0}$)	$2\times2^{27}\times2^{x-12}$	2^{x-16}
10	k_3^0	$k_{73\sim70}$	2 ($k_{71\sim70}$)	$2\times2^{29}\times2^{x-16}$	2^{x-20}
11	k_3^5	$k_{13\sim10}$	4	$2\times2^{33}\times2^{x-20}$	2^{x-24}
12	k_{22}^6	$k_{32\sim22}$	2 ($k_{32\sim31}$)	$2\times2^{35}\times2^{x-24}$	2^{x-28}
13	k_{22}^1	$k_{17\sim0}$	4 (k_{22}^1)	$2\times2^{39}\times2^{x-28}$	2^{x-32}
14	k_{21}^6	$k_{61\sim51}$	4 ($k_{59\sim56}$)	$2\times2^{43}\times2^{x-32}$	2^{x-36}
15	k_{22}^0	$k_{77\sim10}$	4 (k_{22}^0)	$2\times2^{47}\times2^{x-36}$	2^{x-40}
16	k_{22}^7	$k_{39\sim26}$	4 (k_{22}^7)	$2\times2^{51}\times2^{x-40}$	2^{x-44}

Here, we don't fix $k_{79\sim0}$ to a certain machine, so there are only two cases for $k_{54\sim51}$. Once we recover all masterkey bits, we finally check which machine such masterkey belongs to.). The time complexity of this step is $2\times2^2\times2^{x+20}=2^{x+23}$, and the number of pairs remained is $2^{x+20}\times2^{-4}=2^{x+16}$. Similarly, we can handle other subkey nibbles on rounds $1\sim3$ within steps $2\sim11$. Finally, 33 bits of masterkey are guessed in this part, and 2^{x-24} pairs are left.

4.2.4. Filtering on Rounds $21\sim22$. In this part, we guess the subkey nibbles k_{22}^6, k_{22}^1, k_{21}^6, k_{22}^0, k_{22}^7 one by one to further filter pairs, which are summarized within steps $12\sim16$ in Table 12. Luckily, k_{22}^6 can be calculated by the masterkey bits $k_{32\sim22}$ in which there are only 2 unknown bits $k_{32\sim31}$, while k_{21}^6 is exactly the 4 bit masterkey $k_{59\sim56}$. By guessing such subkey nibbles one by one, ciphertext pairs can be decrypted to the end of distinguisher. In the end, about 2^{x-44} pairs are filtered to satisfy the output difference of distinguisher $(00000000, 0E000000)$. Until now, there are 51 bits of key information (including 39 masterkey bits and 12 subkey bits) guessed.

4.2.5. Exhaustive Search of Remained Key Bits. In order to recover the right key with a high probability, we expect there are 2 pairs left under right key, which means $x=29$ in the data collection phase. Once 2 pairs or larger than 2 pairs are left under certain guessed key, we regard such key as a candidate key. Then we search exhaustively remaining key bits and check if it is right with another plaintext/ciphertext pairs.

4.2.6. Complexity Analysis. According to the data collection phase, we can see that the data complexity is 2^{61} chosen plaintexts when we take $x=29$. In this case, there are expected 2 pairs left under tthe right key, which is enough to recover the right key with a high probability. The time complexity mainly happens in steps $1\sim16$ when filtering on rounds $1\sim3$ and $21\sim22$, which are listed in Table 12 one by one step. By summing the time complexities of all steps, the total time is about $2^{45.54}$ 22-round encryptions, which is

$$\left(2^{52}+2\times2^{51}+2\times2^{48}+2^{47}+2\times2^{46}+2^{45}+2\times2^{43}+5\times2^{41}\right)\times\frac{1}{8}\times\frac{1}{22}=2^{45.54}. \tag{2}$$

According to [29], the signal-to-noise ratio is

$$S_N=\frac{(2^\kappa-1)p}{\lambda\gamma-p}, \tag{3}$$

where p denotes the probability of the related-key differential distinguisher, κ denotes number of key bits to recover, λ denotes the probability that a pair survives the filtering, γ denotes keys suggested by each pair surviving the filtering.

Based on signal-to-noise ratio S_N, the success probability of attack is

$$P_s=\phi\left(\frac{\sqrt{\mu S_N}-\phi^{-1}\left(1-2^{-a}\right)}{\sqrt{S_N+1}}\right), \tag{4}$$

where μ denotes that the number of the right pairs, a denotes that a-bit advantage.

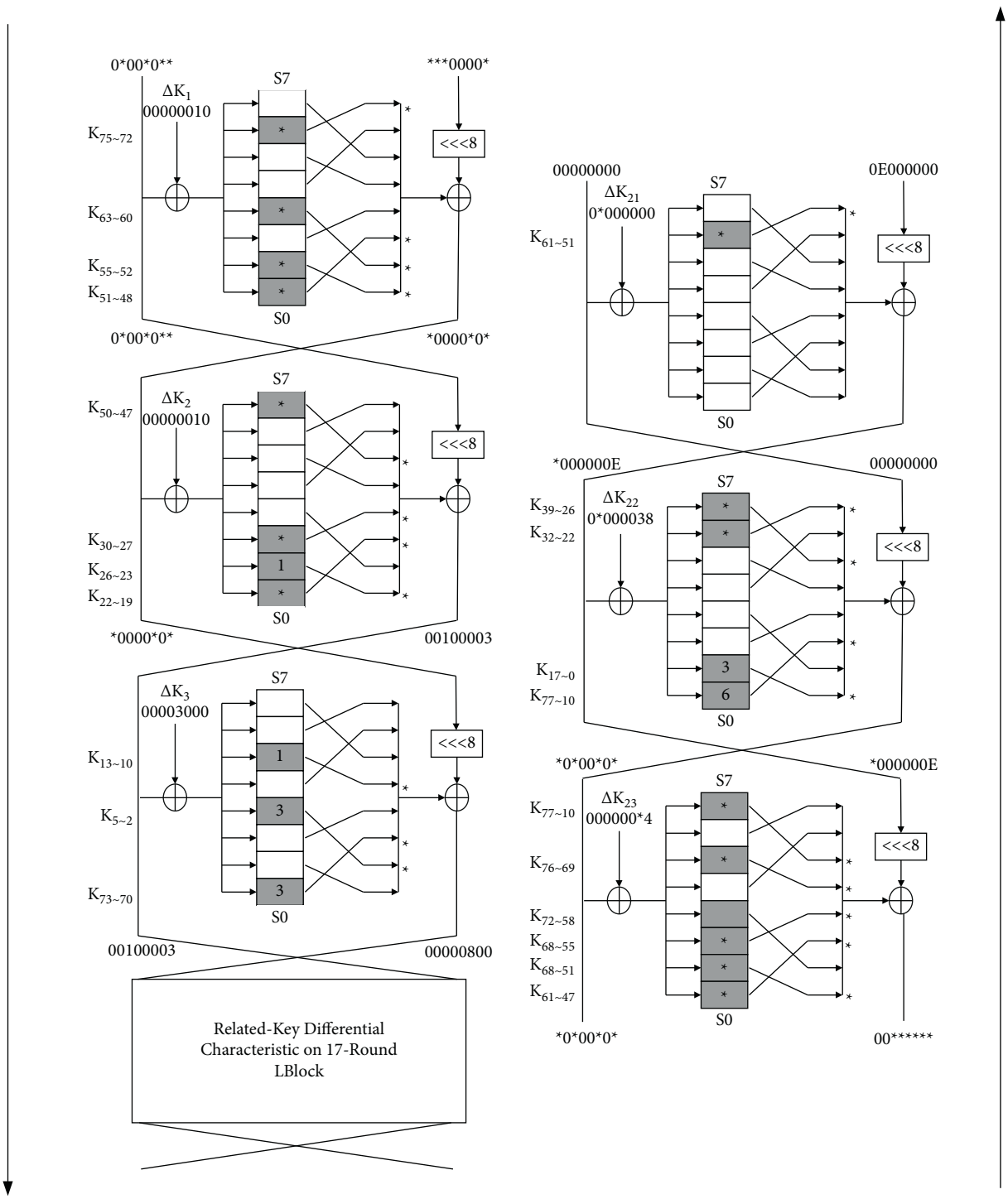

Figure 3: Related-key differential attack on 23-round LBlock.

We can calculate that signal-to-noise ratio is $S_N = (2^{51} - 1)2^{-60}/2^{-44} - 2^{-60} = 2^{35}$, when $\mu = 2, a = 51$. Thus, $P_s = 92.1\%$ according to equation (4).

4.3. Related-Key Differential Attack on 23-Round LBlock. In Subsection 4.1, we search out an expected 17-round related-key differential distinguisher on LBlock, whose total probability is 2^{-70} including 2^{-60} on cipher itself and 2^{-10} on the key schedule. In this subsection, we extend three rounds before this distinguisher and three rounds behind this distinguisher to attack 23-round LBlock. It is shown in Figure 3.

4.3.1. Data Collection. From Figure 3, we can see that the forms of differences on plaintext and ciphertext are $\triangle P = (0 * 00 * 0 * * , * * * 0000 *)$ and

Table 13: Related-key differential attack process on 23-round LBlock.

Step	Subkey	Master Key	#Bits	Time	#Pairs left
1	k_1^1	$k_{55\sim52}$	2 (k_{55}, 2 cases for $k_{54\sim51}$)	$2\times2^2\times2^{x+40}$	2^{x+36}
2	k_1^0	$k_{51\sim48}$	3 ($k_{50\sim48}$)	$2\times2^5\times2^{x+36}$	2^{x+32}
3	k_1^6	$k_{75\sim72}$	4	$2\times2^9\times2^{x+32}$	2^{x+28}
4	k_2^7	$k_{50\sim47}$	1 (k_{47})	$2\times2^{10}\times2^{x+28}$	2^{x+24}
5	k_1^3	$k_{63\sim60}$	4	$2\times2^{14}\times2^{x+24}$	2^{x+20}
6	k_2^1	$k_{26\sim23}$	3 (k_{26}, 4 cases for $k_{25\sim22}$)	$2\times2^{17}\times2^{x+20}$	2^{x+16}
7	k_2^0	$k_{22\sim19}$	3 ($k_{21\sim19}$)	$2\times2^{20}\times2^{x+16}$	2^{x+12}
8	k_2^2	$k_{30\sim27}$	4	$2\times2^{24}\times2^{x+12}$	2^{x+8}
9	k_3^3	$k_{5\sim2}$	3 ($k_{5\sim4}$, 2 cases for $k_{3\sim0}$)	$2\times2^{27}\times2^{x+8}$	2^{x+4}
10	k_3^0	$k_{73\sim70}$	2 ($k_{71\sim70}$)	$2\times2^{29}\times2^{x+4}$	2^x
11	k_3^5	$k_{13\sim10}$	4	$2\times2^{33}\times2^x$	2^{x-4}
12	k_{23}^1	$k_{68\sim51}$	4 (k_{23}^1)	$2\times2^{37}\times2^{x-4}$	2^{x-8}
13	k_{22}^6	$k_{32\sim22}$	2 ($k_{32\sim31}$)	$2\times2^{39}\times2^{x-8}$	2^{x-12}
14	k_{23}^5	$k_{76\sim69}$	2 (k_{76}, k_{69})	$2\times2^{41}\times2^{x-12}$	2^{x-16}
15	k_{23}^7	$k_{77\sim10}$	4 (k_{23}^7)	$2\times2^{45}\times2^{x-16}$	2^{x-20}
16	k_{23}^2	$k_{68\sim55}$	4 (k_{23}^2)	$2\times2^{49}\times2^{x-20}$	2^{x-24}
17	k_{23}^0	$k_{61\sim47}$	4 ($k_{59\sim56}$)	$2\times2^{53}\times2^{x-24}$	2^{x-28}
18	k_{22}^1	$k_{17\sim0}$	4 (k_{22}^1)	$2\times2^{57}\times2^{x-28}$	2^{x-32}
19	k_{21}^6	$k_{61\sim51}$	0	$2\times2^{57}\times2^{x-32}$	2^{x-36}
20	k_{22}^0	$k_{77\sim10}$	4 (k_{22}^0)	$2\times2^{61}\times2^{x-36}$	2^{x-40}
21	$k_{23}^3k_{22}^7$	$k_{72\sim58}k_{39\sim26}$	8 ($k_{23}^3k_{22}^7$)	$2\times2^{69}\times2^{x-40}$	2^{x-44}

$\triangle C = (*0*00*0*, 00******)$ respectively, where $*$ denotes any 4 bit value. According to the form of difference on plaintext, we can build 2^x structures and each structure involves 2^{32} plaintexts. Totally, the data complexity is 2^{x+32} chosen plaintexts. Since each structure composes 2^{64} plaintext pairs, there are expected $2^{x+64-32-60} = 2^{x-28}$ plaintext pairs to satisfy the 17-round distinguisher on both input and output differences.

4.3.2. Filtering on Ciphertext. According to the form of ciphertext difference $\triangle C$, there are on average $2^{x+64-24} = 2^{x+40}$ pairs remained after the filtering on ciphertext.

4.3.3. Filtering on Rounds $1 \sim 3$. In this part, because we are using the same distinguisher as that in 22-round attack, the details are the same as the Filtering on rounds $1 \sim 3$ in Subsection 4.2.

4.3.4. Filtering on Rounds $21 \sim 23$. In this part, we guess the subkey nibbles $k_{23}^1, k_{22}^6, k_{23}^5, k_{23}^7, k_{23}^2, k_{23}^0, k_{22}^1, k_{21}^6, k_{22}^0, k_{23}^3, k_{22}^7$ one by one to further filter pairs, which are summarized within steps $12 \sim 21$ in Table 13. Luckily, k_{22}^6 can be calculated by the masterkey bits $k_{32\sim22}$ in which there are only 2 unknown bits $k_{32\sim31}$, k_{23}^5 can be calculated by the masterkey bits $k_{76\sim69}$ in which there are only 2 unknown bits k_{76} and k_{69}, k_{21}^6 can be calculated by the masterkey bits $k_{61\sim51}$ in which there are not unknown bits, while k_{23}^0 is exactly the 4 bit masterkey $k_{59\sim56}$. By guessing such subkey nibbles one by one, ciphertext pairs can be decrypted to the end of distinguisher. In the end, about 2^{x-44} pairs are filtered to satisfy the output difference of distinguisher $(00000000, 0E000000)$. Until now, there are 65 bits of key information (including 37 masterkey bits and 28 subkey bits) guessed.

4.3.5. Exhaustive Search of Remained Key Bits. In order to recover the right key with a high probability, we expect there are 2 pairs left under right key, which means $x = 29$ in the data collection phase. Once 2 pairs or larger than 2 pairs are left under certain guessed key, we regard such key as a candidate key. Then we search exhaustively remaining key bits and check if it is right with another plaintext/ciphertext pairs.

4.3.6. Complexity Analysis. According to the data collection phase, we can see that the data complexity is 2^{61} chosen plaintexts when we take $x = 29$. In this case, there are expected 2 pairs left under right key, which is enough to recover the right key with a high probability. The time complexity mainly happens in steps $1 \sim 21$ when filtering on rounds $1 \sim 3$ and $21 \sim 23$, which are listed in Table 13 one by one step. By summing the time complexities of all steps, the total time is about $2^{65.48}$ 23-round encryptions, which is

$$\left(2^{72} + 2\times2^{71} + 2\times2^{68} + 2^{67} + 2\times2^{66} + 2^{65} + 2\times2^{63} + 7\times2^{61} + 2\times2^{56}\right)\times\frac{1}{8}\times\frac{1}{23} = 2^{65.48}. \quad (5)$$

According to equations (3) and (4), the signal-to-noise ratio is $S_N = (2^{65} - 1)2^{-60}/2^{-24} - 2^{-60} = 2^{29}$, when $\mu = 2, a = 65$, so the final success rate is $P_s = 92.1\%$.

5. Conclusion

In this paper, we use SMT-based approaches to evaluate the security bounds of LBlock against related-key differential cryptanalysis. Firstly, we propose tighter security bounds, including both the minimal number of active S-boxes and the upper bounds of probabilities for up to 19 rounds of LBlock. As far as we know, these are the best security bounds so far. Secondly, we find a 17-round related-key differential characteristic, whose total probability is 2^{-70} including probability 2^{-60} on the encryption algorithm itself. With this characteristic, we mount key-recovery attacks on 22-round and 23-round LBlock. These are the best related-key differential attacks on LBlock so far. However, there are still some problems that remained in this work that we need to further study. For example, We will focus on how to find as many useful related-key differentials with high probability as possible so as to increase the total probability of the related-key differential distinguisher. With such distinguisher, we can further improve the related-key differential attack on LBlock.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work has been supported by NSFC Projects (No. 61902100), Key Research, Development Program of Zhejiang Province (No. 2020C01078).

References

[1] C. Pei, Y. Xiao, W. Liang, and X. Han, "Trade-off of security and performance of lightweight block ciphers in industrial wireless sensor networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2018, no. 1, p. 117, 2018.

[2] A. Bogdanov, L. R. Knudsen, G. Leander et al., "PRESENT: an ultra-lightweight block cipher," in *Proceedings of the Cryptographic Hardware and Embedded Systems - CHES 2007, 9th International Workshop*, P. Paillier and I. Verbauwhede, Eds., vol. 4727, pp. 450–466, Springer, Vienna, Austria, September 2007.

[3] J. Guo, T. Peyrin, A. Poschmann, and M. J. B. Robshaw, "The LED block cipher," *IACR Cryptol. ePrint Arch.*, p. 600, 2012.

[4] D. Hong, J. Sung, S. Hong et al., "HIGHT: a new block cipher suitable for low-resource device," in *Proceedings of the Cryptographic Hardware and Embedded Systems - CHES 2006, 8th International Workshop*, L. Goubin and M. Matsui, Eds., vol. 4249, pp. 46–59, Springer, Yokohama, Japan, October 2006, Lecture Notes in Computer Science.

[5] W. Wu, L. Zhang, and L. Block, "A lightweight block cipher," Edited by J. López and G. Tsudik, Eds., in *Proceedings of the Applied Cryptography and Network Security - 9th International Conference, ACNS 2011*, vol. 6715, pp. 327–344pp. 327–Lecture Notes in Computer Science, Nerja, Spain, June 2011.

[6] R. Beaulieu, D. Shors, J. Smith, S. Treatman-Clark, B. Weeks, and L. Wingers, "The SIMON and SPECK families of lightweight block ciphers," *IACR Cryptol. ePrint Arch.*, p. 404, 2013.

[7] C. Beierle, J. Jean, S. Kölbl et al., "The SKINNY family of block ciphers and its low-latency variant MANTIS," *IACR Cryptol. ePrint Arch.*vol. 9815, p. 660, 2016.

[8] L. Zhang, W. Wu, Y. Wang, S. Wu, and J. Zhang, "Lac: A lightweight authenticated encryption cipher. Submission to CAESAR," 2014, http://competitions.cr.yp.to/round1/lacv1.pdf.

[9] E. Biham and A. Shamir, "Differential cryptanalysis of des-like cryptosystems," *Journal of Cryptology*, vol. 4, no. 1, pp. 3–72, 1991.

[10] A. D. Dwivedi, P. Morawiecki, and G. Srivastava, "Differential cryptanalysis of round-reduced SPECK suitable for internet of things devices," *IEEE Access*, vol. 7, pp. 16476–16486, 2019.

[11] L. R. Knudsen and D. A. Wagner, "Integral cryptanalysis," in *Proceedings of the Fast Software Encryption, 9th International Workshop, FSE 2002*, J. Daemen and V. Rijmen, Eds., vol. 2365, pp. 112–127, Springer, Leuven, Belgium, February 2002, Lecture Notes in Computer Science.

[12] E. Biham, A. Biryukov, and A. Shamir, "Cryptanalysis of skipjack reduced to 31 rounds using impossible differentials," *Journal of Cryptology*, vol. 18, no. 4, pp. 291–311, 2005.

[13] D. A. Wagner, "The boomerang attack," in *Proceedings of the Fast Software Encryption, 6th International Workshop, FSE '99*, L. R. Knudsen, Ed., vol. 1636, pp. 156–170, Springer, Rome, Italy, March 1999, Lecture Notes in Computer Science.

[14] Y. Liu, D. Gu, Z. Liu, and W. Li, "Impossible differential attacks on reduced-round lblock," in *Proceedings of the Information Security Practice and Experience - 8th International Conference, ISPEC 2012*, M. D. Ryan, B. Smyth, and G. Wang, Eds., vol. 7232, pp. 97–108, Springer, Hangzhou China, April 2012, Lecture Notes in Computer Science.

[15] F. Karakoç, H. Demirci, and A. E. Harmanci, "Impossible differential cryptanalysis of reduced-round lblock," in *Proceedings of the Information Security Theory and Practice. Security, Privacy and Trust in Computing Systems and Ambient Intelligent Ecosystems - 6th IFIP WG 11.2 International Workshop, WISTP 2012*, I. G. Askoxylakis, H. C. Pöhls, and J. Posegga, Eds., vol. 7322, pp. 179–188, Springer, Egham, UK, June 2012.

[16] S. Sun, L. Hu, P. Wang, K. Qiao, X. Ma, and L. Song, P. Sarkar and T. Iwata, Automatic security evaluation and (related-key) differential characteristic search: application to simon, present, lblock, DES(L) and other bit-oriented block ciphers," *Advances in Cryptology - ASIACRYPT 2014 - 20th International Conference on the Theory and Application of Cryptology and Information Security, Kaoshiung, Taiwan*, vol. 8873, Springer, Proceedings, Part I. Lecture Notes in Computer Science, , pp. 158–178, December 2014.

[17] S. Sun, L. Hu, M. Wang et al., *Towards Finding the Best Characteristics of Some Bit-Oriented Block Ciphers and Automatic Enumeration of (Related-key) Differential and Linear*

Characteristics with Predefined Properties, Cryptology ePrint Archive, 2014, https://eprint.iacr.org/2014/747.

[18] M. Minier and M. Naya-Plasencia, "A related key impossible differential attack against 22 rounds of the lightweight block cipher lblock," *Information Processing Letters*, vol. 112, no. 16, pp. 624–629, 2012.

[19] L. Wen, M. Q. Wang, and J. Y. Zhao, "Related-key impossible differential attack on reduced-round lblock," *Journal of Computer Science and Technology*, vol. 29, no. 1, pp. 165–176, 2014.

[20] G. Vijay, H. Trevor, S. Mate, L. Dan, and G. Ryan, "STP," 2014, https://stp.github.io/.

[21] C. Zhou, W. Zhang, T. Ding, and Z. Xiang, "Improving the milp-based security evaluation algorithm against differential/linear cryptanalysis using A divide-and-conquer approach," *IACR Transactions on Symmetric Cryptology*, vol. 2019, no. 4, pp. 438–469, 2020.

[22] J. Chen and A. Miyaji, "Differential cryptanalysis and boomerang cryptanalysis of lblock," in *Proceedings of the Security Engineering and Intelligence Informatics - CD-ARES 2013 Workshops: MoCrySEn and SeCIHD, Regensburg*, A. Cuzzocrea, C. Kittl, D. E. Simos, E. R. Weippl, and L. Xu, Eds., vol. 8128, pp. 1–15, Springer, Germany, September 2013, Proceedings Lecture Notes in Computer Science.

[23] A. Khalesi, H. Bahramgiri, and D. Mansuri, "A new method for accelerating impossible differential cryptanalysis and its application on lblock," *ISC Int. J. Inf. Secur.*vol. 8, no. 1, pp. 73–84, 2016.

[24] S. Liu, Z. Gong, and L. Wang, "Improved related-key differential attacks on reduced-round lblock," in *Proceedings of the Information and Communications Security - 14th International Conference, ICICS 2012*, T. W. Chim and T. H. Yuen, Eds., vol. 7618, pp. 58–69, Springer, Hong Kong, China, October 2012, Proceedings. Lecture Notes in Computer Science.

[25] R. Ankele and S. Kölbl, "Mind the gap - a closer look at the security of block ciphers against differential cryptanalysis," in *Proceedings of the Selected Areas in Cryptography - SAC 2018 - 25th International Conference*, C. Cid and M. J. J. Jr, Eds., vol. 11349, pp. 163–190, Springer, Calgary AB Canada, August 2018, Revised Selected Papers. Lecture Notes in Computer Science.

[26] S. A. Azimi, A. Ranea, M. Salmasizadeh, J. Mohajeri, M. R. Aref, and V. Rijmen, "A bit-vector differential model for the modular addition by a constant and its applications to differential and impossible-differential cryptanalysis," *IACR Cryptol. ePrint Arch. p.*vol. 512, 2022.

[27] S. Kölbl, G. Leander, and T. Tiessen, "Observations on the SIMON block cipher family," in *Proceedings of the Advances in Cryptology - CRYPTO 2015 - 35th Annual Cryptology Conference*, R. Gennaro and M. Robshaw, Eds., vol. 9215, pp. 161–185, Springer, Santa Barbara, CA, USA, August 2015, Proceedings, Part I. Lecture Notes in Computer Science.

[28] Y. Liu, H. Liang, M. Li et al., "STP models of optimal differential and linear trail for s-box based ciphers," *Science China Information Sciences*, vol. 64, no. 5, p. 159103, 2021.

[29] A. A. Selçuk, "On probability of success in linear and differential cryptanalysis," *Journal of Cryptology*, vol. 21, no. 1, pp. 131–147, 2008.