IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received November 10, 2019, accepted November 24, 2019, date of publication December 5, 2019, date of current version December 18, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2957581

Related-Key Differential Cryptanalysis of the Reduced-Round Block Cipher GIFT

MEICHUN CAO AND WENYING ZHANG
School of Information Science and Engineering, Shandong Normal University, Jinan 250014, China

Corresponding author: Wenying Zhang (zhangwenying@sdnu.edu.cn)

This work was supported by the National Natural Science Foundation of China under Grant 61672330 and Grant 61602287.

ABSTRACT GIFT is a lightweight block cipher that was proposed by Banik *et al.* at CHES 2017, which is said to be a direct improvement over PRESENT since "that provides a much increased efficiency in all domains (smaller and faster)" and improves the security weaknesses of the latter. At Asiacrypt in 2014, Sun *et al.* introduced a bit-oriented mixed integer linear programming (MILP) method to search for the differential characteristics of block ciphers. In this paper, we use the differential cryptanalysis method based on this automated tool to analyse GIFT. We propose 12-round and 13-round related-key differential characteristics of GIFT-64 and 7-round and 10-round related-key differential characteristics of GIFT-128. By using them as distinguishers, we apply key recovery attacks on the 19-round and 20-round reduced GIFT-64 with data complexities of 2^{47} and 2^{56} plaintexts, respectively, which mean that the data complexities are lower. Furthermore, we improve the GIFT-64 key recovery attack using differential cryptanalysis by one round over the previous differential cryptanalysis.

INDEX TERMS GIFT block cipher, mixed integer linear programming (MILP), differential cryptanalysis, related-key differential cryptanalysis.

I. INTRODUCTION

The earliest method utilized to protect the integrity and confidentiality of vulnerable data was symmetric encryption, which included block ciphers, stream ciphers, and hash functions. The data encryption standard (DES) that was proposed in the early 1970s and the advanced encryption standard (AES) that was determined in October 2000 are the two most well-known classical encryption algorithms. The DES is insecure due to the 56-bit key size being too small to resist attacks because of the rapidly improving computing capabilities. The AES later replaced the data encryption standard (DES): it has been widely used and has received much attention.

The extensive deployment of tiny computing devices in the internet of things (IoT) and the energy internet is the main trend in this century. These devices are routinely featured in consumer items, and they form an integral part of a pervasive - and unseen - communication infrastructure. It is already being recognized that such deployments bring a range of very particular security risks, such as the privacy leakage and security threats of internet of electric vehicles (IoEV)-based demand response (DR) that was introduced in [1], and the security risks of the vehicle-to-grid (V2G) energy trading in cyber physical systems that was mentioned in [2]. In [3], the state-of-the-art lightweight encryption technologies that are necessary for IoT-based applications, which can be effectively implemented in constrained devices, are outlined. In addition, the article also addressed some of the available lightweight block ciphers and they were compared. The PRESENT [4]–[6] lightweight block cipher that was proposed at CHES 2007 is one cryptographic primitive that provides a cryptographic solution for IoT security. It is a hardware-optimized block cipher that has been carefully designed with area and power constraints to enhance the security of the IoT. GIFT [7] is an improved version of the PRESENT block cipher, and it was designed by the same team as the PRESENT. GIFT is more efficient in all domains (smaller and faster) while correcting the well-known weakness of the PRESENT with respect to linear hulls. Therefore, it is one of the most energy efficient ciphers today. The designers claimed that it provides strong bounds with regards to differential/linear attacks.

Differential cryptanalysis [8] assesses a chosen-plaintext attack and studies the influence of a pair of plaintext differences on the output differences of the subsequent rounds

The associate editor coordinating the review of this manuscript and approving it for publication was Zhenyu Zhou.

in an iterative cipher. Due to its generality, differential cryptanalysis is a cryptanalysis tool that can be widely used in a large variety of encryption algorithms [9], and it can also be used to define new attack methods. The resistance to differential cryptanalysis is one of the basic criteria to evaluate the security of block encryption algorithms. Related key attacks [10], [11] allow cryptographers to obtain plaintext/ciphertext pairs by using different keys. The attacker knows some relations between the keys, but the actual values of the keys are unknown. In addition, the relationships between the keys can be selected by the attacker. In a related-key differential attack model, it is possible to cancel internal state differences by using corresponding key differences, which makes the internal state differences to spread more slowly. For example, when the difference of the corresponding positions in the exclusive OR operations of the internal state difference and the round key difference is 1, the difference of the calculation result is 0. This finding creates a higher probability differential characteristic and can cover more rounds than the single key attack, which may result in greatly improved time complexity. As far as we know, there are few studies that evaluate the cipher in the related-key model. Notice that the key schedule of the GIFT cipher is linear; therefore, the attacks under the related-key setting may penetrate more rounds, reduce the attack complexity by cancelling the key difference and the state difference, and reveal a better picture of its security. However, there is no known related-key differential attack for GIFT.

It has been found that many classical cryptanalysis methods, including differential cryptanalysis and linear attacks, can be transformed into mathematical optimization problems in order to obtain the minimum or maximum value of the objective function under certain constraints. Mixed integer linear programming (MILP) is a method that is often used to solve optimization problems in business and economics. MILP-based cryptanalysis techniques greatly reduce the workloads of designers and cryptanalysts because it only involves constructing simple equations that are input into the MILP solvers. Since only a small amount of programming is required, the time that is spent on cryptanalysis and the possibility of human errors are significantly reduced. One of MILP's most successful applications so far is searching for differential paths. Mouha *et al.* applied the MILP method to calculate the active S-boxes of the word-based block cipher [12]. Sun *et al.* [6] established an MILP model for bit-oriented block ciphers based on Mouha *et al.*'s algorithm [12]. Xiang *et al.* proposed an MILP model for searching for integral distinguishers [5]. Zhang and Rijmen proposed an accurate description of the division property of block ciphers by using a binary linear layer [13]. Hao *et al.* and Wang *et al.* independently presented cube attacks on the stream ciphers based on MILP [14], [15]. Huang *et al.*, Song *et al.*, and Li *et al.* also used MILP key recovery attacks on Keccak-MAC and Keyak [16]–[18].

Related work. Shortly after the proposal of the GIFT, Zhu *et al.* proposed a differential attack for the 19-round GIFT-64 based on a 12-round differential distinguisher under the single-key setting [19]. In addition, the security of the cipher against meet-in-the-middle (MITM) attacks was studied in [20]. Liu and Sasaki found 19-round related-key boomerang distinguishers in the GIFT-64 and GIFT-128 lightweight block ciphers [21]. In [22], the combination of side-channel analysis (SCA) and the differential fault attack (DFA) can respectively recover the 32 bits and 64 bits of the last round keys of GIFT-64 and GIFT-128 by approximating 9-18 fault injections and 6-9 fault injections in the average cases.

Our contribution. In this paper, by using an MILP model, we examine the security of GIFT's resistance to related-key differential attacks. Although the designers of GIFT do not claim related-key security, they evaluated its effectiveness against related-key differential attacks. The best bit positions of the key addition and 16-bit rotations were chosen to optimize the related-key differential bounds. Therefore, the designers did their best to resist related-key differential attacks. However, in this paper, we present 12-round and 13-round GIFT-64 related-key differential characteristics with probabilities of 2^{-37} and $2^{-47.83}$, respectively. In addition, we also present the related-key differential characteristics for the 7-round and 10-round GIFT-128 with probabilities of $2^{-15.83}$ and $2^{-72.66}$, respectively. These probabilities are higher than previous results. We propose key-recovery attacks by using these related-key differential characteristics. The data complexities of the attacks are 2^{47} and 2^{56} for the 19-round reduced GIFT-64 and the 20-round reduced GIFT-64, respectively, which are lower than the complexities that were given in previous works. Our key recovery attack on the 20-round reduced GIFT-64 improves one more round than the results in [19].

A summary of the comparisons of our results with the previous cryptanalysis techniques on GIFT is shown in Table 1, where MITM, RK-B, and RKD stand for meet-in-the-middle, related-key boomerang and related-key differential, respectively.

TABLE 1. Summary of cryptography analysis on GIFT.

Algorithm	Type	Attack #rd	Data	Source
GIFT-64	Integral	14	-	[7]
	Differential	19	2^{63}	[19]
	MITM	15	-	[20]
	RK-B	23	$2^{63.3}$	[21]
	RKD	19	2^{47}	Sect. IV
	RKD	20	2^{56}	Sect. IV
GIFT-128	Differential	23	2^{120}	[19]
	RK-B	21	$2^{126.6}$	[21]

Organization. The rest of the paper is organized as follows. Section 2 gives the preliminaries of GIFT and the interpretations of the automatic differential analysis. It is followed by an algorithm that searches for the related-key differentials with the highest probabilities in Section 3. In Section 4, we apply our new related-key differentials to the key recovery

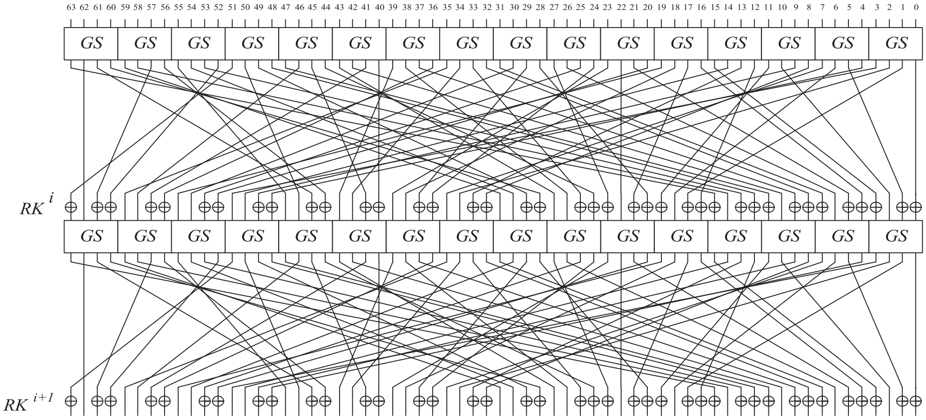

FIGURE 1. The structure of GIFT-64.

TABLE 2. Specifications of GIFT S-box *GS*.

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

attack by using GIFT. Finally, we conclude the paper in Section 5.

II. PRELIMINARIES

A. DESCRIPTION OF GIFT

GIFT is a substitution-permutation network (SPN) block cipher. It has two versions, GIFT-64 and GIFT-128, both of which have 128-bit secret key length. The former has a block size of 64 bits and 28 iteration rounds, while the latter has a block size of 128 bits and 40 iteration rounds. The round function has three operations: SubCells, PermBits and AddRoundKey. Fig. 1 shows the structure of the round function of GIFT-64. Similarly, GIFT-128 adopts 32 4-bit S-boxes for each round. The symbol "$\oplus$" indicates the exclusive OR, which is abbreviated to XOR. For example, there are two bits a and b for the XOR operation. Then, if the values of a and b are not the same, the XOR result is 1; otherwise, the XOR result is 0. If there are some vectors that are XORed, the bits of the corresponding positions of these vectors are XORed according to the above rule, that is, bitwise XOR.

Initialization. The cipher accepts an n-bit state $b_{n-1}b_{n-2}\ldots b_1b_0$ as the internal state S, where n is equal to 64 or 128, corresponding to GIFT-64 or GIFT-128, respectively. Note that b_{n-1} is the most significant bit.

SubCells. The internal state of this operation can be represented as $S = w_{s-1}w_{s-2}\ldots w_1w_0$, where $s = 16, 32$; and w_i, $i \in \{0, \ldots, s-1\}$, are 4-bit nibbles. The nonlinear mapping applies the S-box parallel to each nibble of the internal state.

$$w_i \leftarrow GS(w_i), \forall i \in \{0, \ldots, s-1\}. \quad (1)$$

Both versions of GIFT apply the same invertible 4-bit S-box, GS. The truth-table for the S-box in hexadecimal notation is shown in Table 2.

PermBits. This operation maps the bits of the cipher state S from position i to $P(i)$. The bit permutation specifications of GIFT-64 and GIFT-128 can be found in the specification of the cipher [7].

AddRoundKey. The $n/2$ bit round key is XORed with part of the internal state bits. The round key is extracted from the 128-bit key register K as $RK = U \parallel V = u_s, \ldots, u_1, u_0 \parallel v_s, \ldots, v_1, v_0$, where $s = 16$ and 32 correspond to GIFT-64 and GIFT-128, respectively. The symbol "$\parallel$" represents a concatenation between vectors (for example, $(0010) \parallel (1110) = (00101110)$). Then, a part of the internal state bits are XORed with RK as follows.

For GIFT-64, the internal state bits $\{b_{4i+1}\}$ and $\{b_{4i}\}$ are XORed with U and V, respectively.

$$\begin{cases} U \leftarrow k_1, V \leftarrow k_0, \\ b_{4i+1} \leftarrow b_{4i+1} \oplus u_i, \; b_{4i} \leftarrow b_{4i} \oplus v_i, \\ \forall i \in \{0, \ldots, 15\}. \end{cases} \quad (2)$$

For GIFT-128, the cipher states $\{b_{4i+2}\}$ and $\{b_{4i+1}\}$ are XORed with U and V, respectively.

$$\begin{cases} U \leftarrow k_5 \parallel k_4, V \leftarrow k_1 \parallel k_0. \\ b_{4i+2} \leftarrow b_{4i+2} \oplus u_i, \; b_{4i+1} \leftarrow b_{4i+1} \oplus v_i, \\ \forall i \in \{0, \ldots, 31\}. \end{cases} \quad (3)$$

After AddRoundKey, the 128-bit key state for both versions is then updated as follows.

$$k_7 \parallel k_6 \parallel \ldots \parallel k_1 \parallel k_0 \leftarrow k_1 \ggg 2 \parallel k_0 \ggg 12 \ldots \parallel k_3 \parallel k_2, \quad (4)$$

where "$\ggg i$" is an i bit right rotation within a 16-bit word.

Round Constants. A 6-bit round constant $RC = \{rc_5, rc_4, rc_3, rc_2, rc_1, rc_0\}$ and a single bit "1" are XORed into the cipher state as defined below:

$$\begin{cases} b_{n-1} \leftarrow b_{n-1} \oplus 1, \\ b_{23} \leftarrow b_{23} \oplus rc_5, b_{19} \leftarrow b_{19} \oplus rc_4, b_{15} \leftarrow b_{15} \oplus rc_3, \\ b_{11} \leftarrow b_{11} \oplus rc_2, b_7 \leftarrow b_7 \oplus rc_1, b_3 \leftarrow b_3 \oplus rc_0. \end{cases} \quad (5)$$

where $n-1$, 23, 19, 15, 11, 7 and 3 denote bit positions in the internal state S.

The round constant $RC = \{rc_5, rc_4, rc_3, rc_2, rc_1, rc_0\}$ is initialized to "0", and it is updated before each round as follows:

$$(rc_5, rc_4, rc_3, rc_2, rc_1, rc_0) \leftarrow (rc_4, rc_3, rc_2, rc_1, rc_0, rc_0 \oplus rc_4 \oplus 1). \quad (6)$$

B. AUTOMATIC DIFFERENTIAL ANALYSIS OF BIT-ORIENTED BLOCK CIPHERS

In this section, we describe the MILP-based cryptanalysis methods that were proposed by Mouha *et al.* and Sun *et al.* Mouha *et al.* [12] first proposed the MILP model to calculate the number of active S-boxes in the word-oriented block ciphers propagation process. At Asiacrypt 2014, Sun *et al.* [6] established an MILP model for bit-oriented block ciphers based on Mouha et al.'s algorithm [12]. We briefly recall Sun et al.'s framework. For more details of their framework, we refer the readers to [6], [23].

Definition 1: For each input and output bit-wise difference, we consider a 0-1 variable x_i to denote whether this bit has a nonzero difference or not. Then, the differential vector $x = (x_0, x_1, \ldots, x_{n-1})$ is defined as follows:

$$x_i = \begin{cases} 1, & \text{for nonzero difference at this bit,} \\ 0, & \text{otherwise.} \end{cases} \quad (7)$$

Constraints Describing the XOR Operation. Assuming that the input differences for the XOR operation at the bit-level are x_{in1} and x_{in2} and the corresponding bit-level output difference is x_{out}, the constraints include

$$\begin{cases} x_{in1} + x_{in2} + x_{out} \geq 2d_{\oplus} \\ x_{in1} + x_{in2} + x_{out} \leq 2 \\ d_{\oplus} \geq x_{in1}, \ d_{\oplus} \geq x_{in2}, \ d_{\oplus} \geq x_{out} \end{cases} \quad (8)$$

where $d_{\oplus}$ is a dummy bit variable.

Constraints Describing the S-box Operation. Let $(x_0, \ldots, x_{w-1})$ and $(y_0, \ldots, y_{v-1})$ represent the input and output bit-level differences of a $w \times v$ S-box. Then, we have

$$\begin{cases} A_t - x_i \geq 0, \ i \in \{0, 1, \ldots, w-1\} \\ \sum_{j=0}^{w-1} x_j - A_t \geq 0 \end{cases} \quad (9)$$

where $A_t \in \{0, 1\}$ is a dummy variable that describes whether the S-box is active or not. $A = 1$ holds if and only if $x_0, x_1, \ldots, x_{w-1}$ are not all zero.

The non-zero input difference of the bijective S-boxes must result in a nonzero output difference and vice versa.

H-representation of a convex hull. The convex hull of a set X is the minimum convex set that contains X, where X is a set of discrete points in $\mathbb{R}^n$. A convex hull in $\mathbb{R}^n$ can be described as the common solutions of a system of linear equalities and inequalities. If we treat all m differential patterns of a $w \times v$ S-box as m different discrete points in $\mathbb{R}^{w+v}$, then $(x, y) = (x_0, \ldots, x_{w-1}, y_0, \ldots, y_{v-1})$ can represent a differential pattern of a $w \times v$ S-box, in which x and y are the input and output differences vectors, respectively. The set containing all possible differential patterns of an S-box can be obtained through the difference distribution table (DDT) of the S-box, that is, the input and output differences corresponding to all non-zero entries in the table. As a result, we can describe this finite set with the following inequalities:

$$\begin{cases} wy_{j_0} + wy_{j_1} + \ldots + wy_{j_{v-1}} - (x_{i_0} + x_{i_1} + x_{i_{w-1}}) \geq 0 \\ vx_{i_0} + vx_{i_1} + \ldots + vx_{i_{w-1}} - (y_{j_0} + y_{j_1} + y_{j_{v-1}}) \geq 0 \end{cases} \quad (10)$$

This is called the H-representation of a $w \times v$ S-box. By using the SAGE [24] computer algebra system, many linear inequalities can be obtained for the differential patterns of the S-box. In general, the number of linear inequalities in the convex hull of a set $X \subset \mathbb{R}^n$ increases rapidly as n increases, and as the number of linear inequalities increases, the efficiency of the MILP model decreases dramatically. A 4×4 S-box contains hundreds of inequalities that would be hard to solve in a practical amount of time if all of them are added to the MILP model. To overcome this issue, Sun *et al.* utilized a greedy algorithm to select some "good" inequalities from the convex hull [6].

In [25], Sasaki *et al.* proposed an MILP-based reduction algorithm to extract the optimal combination with the minimal number of linear inequalities from hundreds of inequalities in the H-representation of the convex hull, which removes all the impossible differential patterns of the S-box. The algorithm considers that every impossible pattern in the DDT of an S-box should be excluded from the solution space by at least one linear inequality. Under these constraints, the number of inequalities can be minimized using the MILP model.

C. NOTATIONS

κ_i	The i-th bit of the master key.
$\tilde{\kappa}_i$	The key bit that needs to be guessed.

$$\begin{cases} (p2, p1, p0) = (0, 0, 0), & \text{if } \Pr s[(x3, x2, x1, x0) \to (y3, y2, y1, y0)] = 1 \\ (p2, p1, p0) = (1, 0, 0), & \text{if } \Pr s[(x3, x2, x1, x0) \to (y3, y2, y1, y0)] = 2^{-1.415} \\ (p2, p1, p0) = (0, 1, 0), & \text{if } \Pr s[(x3, x2, x1, x0) \to (y3, y2, y1, y0)] = 2^{-2} \\ (p2, p1, p0) = (0, 0, 1), & \text{if } vs[(x3, x2, x1, x0) \to (y3, y2, y1, y0)] = 2^{-3} \end{cases} \quad (11)$$

P	Plaintext.
C	Ciphertext.
R	Round.
$\Delta SB^{i}_{in/out}$	The input/output difference of S-box of the i-th round.
$\Delta LP^{i}_{in/out}$	The input/output difference of Permutation of the i-th round.
$\Delta AK^{i}_{in/out}$	The input/output difference of AddRoundKey of the i-th round.

III. THE ALGORITHM FOR SEARCHING FOR THE RELATED-KEY DIFFERENTIAL CHARACTERISTICS OF GIFT

A. MILP-BASED MODEL TO SEARCH FOR THE RELATED-KEY DIFFERENTIAL CHARACTERISTICS FOR GIFT

This section describes the high-probability related-key differential paths that are found based on the MILP method. In [23], Sun *et al.* introduced the differential distribution probability of the S-box into the constraints of the MILP model.

We first calculate the H-presentation of the convex hull of differential patterns by using the probabilities that are based on the DDT of the S-box, as shown in Table 3, where α and β represent the input difference and the output difference of the S-box, respectively. The S-box of GIFT has four possible probabilities: $1, 6/16, 4/16$, and $2/16$, i.e., $1, 2^{-1.415}, 2^{-2}$, and 2^{-3}, respectively. For every possible differential pattern $(x_3, x_2, x_1, x_0, y_3, y_2, y_1, y_0)$, we need three extra bits (p_2, p_1, p_0) to encode the differential probability. The corresponding differential pattern with the probability is $(x_3, x_2, x_1, x_0, y_3, y_2, y_1, y_0, p_2, p_1, p_0) \in \mathbb{F}_2^{8+2}$, which satisfies the rules that are listed in (11), as shown at the bottom of the previous page. Then, we apply this rule to generate a set of linear inequalities by using the SAGE [24] computer algebra system. Furthermore, this set can be reduced by using the optimization algorithm from [6], [25]. The inequalities in (12), as shown at the bottom of this page are the 25 inequalities that are used to describe the DDT of the GIFT S-box.

For the bit-oriented block cipher GIFT, supposing that the input difference of the XOR operation is (x_1, x_2) and the corresponding output difference is y, then we have the inequalities in (13) that describe the XOR

$$\begin{cases}
-p0 - p1 - p2 >= -1\\
x3 + 3x2 + 2x1 + 4x0 + 4y3 + 2y2 + 2y1 + 3y0 - 10p0 - 6p1 - 8p2 >= 0\\
-2x3 - x2 - x1 - x0 - 4y3 - y2 + y1 + p0 + 6p1 + 9p2 >= 0\\
4x3 + 3x2 + 3x1 - 2x0 + 4y2 + 3y1 + 2y0 - 11p0 - 9p1 + 2p2 >= 0\\
-2x3 - 4x2 - 2x0 + 4y3 - y2 - 3y1 - 3y0 + 11p0 + 9p1 + 11p2 >= 0\\
x3 - x2 - x1 + 3x0 - 4y3 - 3y1 - 4y0 + 9p0 + 10p1 + 9p2 >= 0\\
x3 + x2 - x1 + 3x0 + 4y3 + y2 + 2y1 + 2y0 - 4p0 - 2p1 - 5p2 >= 0\\
5x3 + x2 + 7x1 + 5x0 - y3 - y2 - 3y1 + 3y0 - 3p1 - p2 >= 0\\
-3x3 + 3x2 - 3x1 - 2x0 - y3 - 3y2 + y1 - 2y0 + p0 + 8p1 + 12p2 >= 0\\
2x2 + x1 + 3x0 + 2y3 + y1 + y0 - 4p0 - 2p1 - 4p2 >= 0\\
-4x3 + x2 - 2x1 - 3x0 - y3 + 2y2 - 2y1 + y0 + 2p0 + 6p1 + 10p2 >= 0\\
4x3 - 4x2 - x1 - x0 + 2y3 - 4y2 - y1 + 2y0 + 7p0 + 6p1 + 8p2 >= 0\\
x3 - x2 - 2x0 + y3 - 2y1 - 2y0 + 5p0 + 6p1 + 5p2 >= 0\\
-x3 - x2 - x0 + y1 + y0 - p0 + 3p1 + 2p2 >= 0\\
-2x3 - 2x2 + 3x1 - 2x0 - 3y3 + 3y2 + 3y1 - y0 - 3p0 + 3p1 + 7p2 >= 0\\
-x3 - x2 - x1 + x0 - y3 - y1 + 3p0 + 3p1 + 4p2 >= 0\\
-x3 + x2 + 2x1 - x0 - 2y3 - 2y2 - 2y1 + y0 + 2p0 + 6p1 + 6p2 >= 0\\
3x3 + x2 - x1 - x0 + 2y3 + 3y2 + y1 - y0 - p1 >= 0\\
2x3 - 2x2 - 5x1 - x0 - y3 - 5y2 - y1 - y0 + 9p0 + 11p1 + 14p2 >= 0\\
-x3 + x2 + 2x1 + 2x0 - y3 - y1 - 2y0 + 2p0 + 5p1 + 2p2 >= 0\\
-x3 + y3 - y2 + y1 - y0 + p0 + 2p1 + 2p2 >= 0\\
x3 + x0 + y3 + y1 + y0 - 2p0 - p1 - 2p2 >= 0\\
4x3 - x2 + 4x1 + 2x0 - y3 + y2 - y1 + 2y0 - p0 - 2p1 >= 0\\
x3 + x2 + x1 - y3 - y2 + y1 - y0 + p1 + 2p2 >= 0\\
-x3 + x2 - 2x1 - 2x0 - y3 + 2y2 - 2y1 + y0 + 2p0 + 5p1 + 6p2 >= 0\\
p_i, x_i, y_i \quad \textit{are binaries}
\end{cases} \tag{12}$$

TABLE 3. Difference distribution table (DDT) of the GIFT S-box.

α	β															
	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	2	2	0	2	2	2	2	2	0	0	2
2	0	0	0	0	0	4	4	0	0	2	2	0	0	2	2	0
3	0	0	0	0	0	2	2	0	2	0	0	2	2	2	2	2
4	0	0	0	2	0	4	0	6	0	2	0	0	0	2	0	0
5	0	0	2	0	0	2	0	0	2	0	0	0	2	2	2	4
6	0	0	4	6	0	0	0	2	0	0	2	0	0	0	2	0
7	0	0	2	0	0	2	0	0	2	2	2	4	2	0	0	0
8	0	0	0	4	0	0	0	4	0	0	0	4	0	0	0	4
9	0	2	0	2	0	0	2	2	2	0	2	0	2	2	0	0
a	0	4	0	0	0	0	4	0	0	2	2	0	0	2	2	0
b	0	2	0	2	0	0	2	2	2	2	0	0	2	0	2	0
c	0	0	4	0	4	0	0	0	2	0	2	0	2	0	2	0
d	0	2	2	0	4	0	0	0	0	0	2	2	0	2	0	2
e	0	4	0	0	4	0	0	0	2	2	0	0	2	2	0	0
f	0	2	2	0	4	0	0	0	0	2	0	2	0	0	2	2

operation.

$$\begin{cases} x_1 + x_2 - y >= 0 \\ x_1 - x_2 + y >= 0 \\ -x_1 + x_2 + y >= 0 \\ x_1 + x_2 + y <= 2 \\ x_i, \quad y \text{ are binaries} \end{cases} \tag{13}$$

The objective function is chosen to minimize $\sum(1415p_2 + 2000p_1 + 3000p_0)$. Now, the MILP model that is constructed by the above techniques utilizing Algorithm 1 can generate the optimal solution corresponding to the characteristic with the maximum probability.

Algorithm 1 Related-Key Differential Characteristic Search Algorithm Based on MILP

Require: r cipher rounds, specific assignment to the difference of the core key k^0

Ensure: differential characteristic with maximal probability

```
1: Establish an empty MILP model $M$.
2: Set $x$, $y$ and $z$ as the input and output of the S-box and the output of the linear layer, respectively.
3: Set $k$ as the key and $s$ as the probability of the DDT.
4: Update $M$ according to the differential propagation rule of the round function.
5: Extra constraint $M.con \leftarrow \sum_{k=0}^{m-1} k_i \geq 1$ for all $i \in \{0, \dots, m-1\}$.
6: Set the objective function $M.obj \leftarrow \sum_{min}(1415p_2 + 2000p_1 + 3000p_0)$.
7: Solve model $M$ using an MILP optimizer.
8: A feasible solution is found in $M$, and save it to a file.
```

We implement Algorithm 1 to search for the related-key differential characteristics. Here, we describe the 13-round characteristic of GIFT-64 with a probability of $2^{-47.83}$ in Table 4 in this section. Due to space limitations, the details of the 12-round characteristic of GIFT-64 and the 7-round and 10-round characteristics of GIFT-128 are respectively shown in Tables 7, 8 and 9 in Appendix A.

TABLE 4. The 13-round related-key differential characteristic of GIFT-64 with a probability of $2^{-47.83}$.

R	Input Difference	Subkey Difference	Probability
1	0000 0000 1C00 0000	0040 1000 0000 0000 0000 9000 0000 0000	
2	0060 0000 0000 0080	0000 0000 0040 1000 0000 0000 0000 9000	2^{-5}
3	0000 0000 0000 2002	0000 0009 0000 0000 0040 1000 0000 0000	$2^{-8.415}$
4	0000 0006 0000 0009	0000 0000 0000 0009 0000 0000 0040 1000	$2^{-13.415}$
5	0009 0000 0000 0000	0010 0001 0000 0000 0000 0009 0000 0000	$2^{-18.415}$
6	0000 0000 0000 1000	0000 0000 0010 0001 0000 0000 0000 0009	$2^{-21.415}$
7	0000 0000 0000 1009	0000 0090 0000 0000 0010 0001 0000 0000	$2^{-24.415}$
8	0000 0000 0000 0009	0000 0000 0000 0090 0000 0000 0010 0001	$2^{-30.415}$
9	0000 0000 0000 0000	0004 0010 0000 0000 0000 0090 0000 0000	$2^{-33.415}$
10	0000 0000 0000 0000	0000 0000 0004 0010 0000 0000 0000 0090	$2^{-33.415}$
11	0000 0000 1001 0000	0000 0900 0000 0000 0004 0010 0000 0000	$2^{-33.415}$
12	00C0 0060 0000 0000	0000 0000 0000 0900 0000 0000 0004 0010	$2^{-39.415}$
13	0000 0000 4001 0000	0001 0100 0000 0000 0000 0900 0000 0000	$2^{-43.415}$
14	0040 0060 0030 0000	-	$2^{-47.83}$

IV. RELATED-KEY DIFFERENTIAL ATTACKS ON GIFT

In this section, we first use the related-key differential characteristic that was introduced in Table 4 to extend the analysis 3 rounds forward and 4 rounds backward to establish the 20-round GIFT-64 key recovery attack, as shown in Table 5. Assume that the rounds in the 20-round reduced GIFT-64 are numbered from round 1 to round 20. Accordingly, the distinguisher that is used in the attack is from round 4 to round 16.

TABLE 5. The 20-round attack on GIFT-64.

State	Difference
$P = \Delta SB_{in}^{1}$	???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ????
ΔLP_{in}^{1}	0??? ?0?? ??0? ???0 0??? ?0?? ??0? ???0 0??? ?0?? ??0? ???0 0??? 10?? ?10? ???0
ΔAK_{in}^{1}	???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? 11?? 0000 0000 0000 0000
ΔSB_{in}^{2}	???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? 11?? 0000 0000 0000 0000
ΔLP_{in}^{2}	??00 01?0 00?? 100? 10?0 0?0? ?0?0 0?0? 00?0 000? 1000 0100 0000 0000 0000 0000
ΔAK_{in}^{2}	1??? 0000 0000 0000 0000 ???? 11?? 0000 0000 0000 0000 0000 ?1?? 1??? 0000 0000
ΔSB_{in}^{3}	1??? 0000 0000 0000 0000 ???? 11?? 0000 0000 0000 0000 0000 ?1?? 1??? 0000 0000
ΔLP_{in}^{3}	0001 0000 0000 0000 0000 1000 0100 0000 0000 0000 0000 0000 0010 0001 0000 0000
ΔAK_{in}^{3}	0000 0000 0000 0000 0000 0000 0000 0011 0001 1100 0000 0000 0000 0000 0000 0000
ΔSB_{in}^{4}	0000 0000 0000 0000 0000 0000 0000 0000 0001 1100 0000 0000 0000 0000 0000 0000
...	...
ΔAK_{out}^{16}	0000 0000 0100 0000 0000 0000 0110 0000 0000 0000 0011 0000 0000 0000 0000 0000
ΔSB_{out}^{17}	0000 0000 ???? 0000 0000 0000 ??1? 0000 0000 0000 ???? 0000 0000 0000 0000 0000
ΔLP_{out}^{17}	000? 000? 000? 0000 ?000 ?000 ?000 0000 0?00 0?00 0?00 0000 00?0 0010 00?0 0000
ΔAK_{out}^{17}	000? 000? 000? 0000 ?001 ?000 ?000 0001 0?00 0?00 0?00 0000 00?0 0010 00?0 0000
ΔSB_{out}^{18}	???? ???? ???? 0000 ???? ???? ???? ???? ???? ???? ???? 0000 ???? ???? ???? 0000
ΔLP_{out}^{18}	0??? ???? 0??? 0??? ?0?? ???? ?0?? ?0?? ??0? ???? ??0? ??0? ???0 ???? ???0 ???0
ΔAK_{out}^{18}	0??? ???? 0??? 0??? ?0?? ???? ?0?? ?0?? ??0? ???? ??0? ??0? ???0 ???? ???0 ???0
ΔSB_{out}^{19}	???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ????
ΔLP_{out}^{19}	???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ????
ΔAK_{out}^{19}	???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ????
ΔSB_{in}^{20}	???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ????
ΔLP_{in}^{20}	???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ????
$C = \Delta AK_{in}^{20}$	???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ????

TABLE 6. The round keys that are used in the 20-round attack on GIFT-64.

R	Key Bits
1	$\kappa\tilde{31}, \kappa\tilde{30}, \kappa\tilde{29}, \kappa\tilde{28}, \kappa\tilde{27}, \kappa\tilde{26}, \kappa\tilde{25}, \kappa\tilde{24}, \kappa\tilde{23}, \kappa\tilde{22}, \kappa\tilde{21}, \kappa\tilde{20}, \kappa 19, \kappa 18, \kappa 17, \kappa 16,$ $\kappa\tilde{15}, \kappa\tilde{14}, \kappa\tilde{13}, \kappa\tilde{12}, \kappa\tilde{11}, \kappa\tilde{10}, \kappa\tilde{9}, \kappa\tilde{8}, \kappa\tilde{7}, \kappa\tilde{6}, \kappa\tilde{5}, \kappa\tilde{4}, \kappa 3, \kappa 2, \kappa 1, \kappa 0,$
2	$\kappa\tilde{63}, \kappa 62, \kappa 61, \kappa 60, \kappa 59, \kappa\tilde{58}, \kappa\tilde{57}, \kappa 56, \kappa 55, \kappa 54, \kappa 53, \kappa 52, \kappa\tilde{51}, \kappa\tilde{50}, \kappa 49, \kappa 48,$ $\kappa\tilde{47}, \kappa 46, \kappa 45, \kappa 44, \kappa 43, \kappa\tilde{42}, \kappa\tilde{41}, \kappa 40, \kappa 39, \kappa 38, \kappa 37, \kappa 36, \kappa\tilde{35}, \kappa\tilde{34}, \kappa 33, \kappa 32,$
3	$\kappa 95, \kappa 94, \kappa 93, \kappa 92, \kappa 91, \kappa 90, \kappa 89, \kappa 88, \kappa 87, \kappa 86, \kappa 85, \kappa 84, \kappa 83, \kappa 82, \kappa 81, \kappa 80,$ $\kappa 79, \kappa 78, \kappa 77, \kappa 76, \kappa 75, \kappa 74, \kappa 73, \kappa 72, \kappa 71, \kappa 70, \kappa 69, \kappa 68, \kappa 67, \kappa 66, \kappa 65, \kappa 64,$
17	$\kappa 23, \kappa 22, \kappa 21, \kappa 20, \kappa 19, \kappa 18, \kappa 17, \kappa 16, \kappa 31, \kappa 30, \kappa 29, \kappa 28, \kappa\tilde{27}, \kappa 26, \kappa\tilde{25}, \kappa 24,$ $\kappa\tilde{15}, \kappa\tilde{14}, \kappa\tilde{13}, \kappa 12, \kappa 11, \kappa 10, \kappa 9, \kappa 8, \kappa 7, \kappa 6, \kappa 5, \kappa 4, \kappa 3, \kappa 2, \kappa 1, \kappa 0,$
18	$\kappa\tilde{55}, \kappa\tilde{54}, \kappa\tilde{53}, \kappa\tilde{52}, \kappa\tilde{51}, \kappa\tilde{50}, \kappa\tilde{49}, \kappa\tilde{48}, \kappa 63, \kappa\tilde{62}, \kappa 61, \kappa 60, \kappa\tilde{59}, \kappa\tilde{58}, \kappa\tilde{57}, \kappa\tilde{56},$ $\kappa\tilde{47}, \kappa\tilde{46}, \kappa\tilde{45}, \kappa\tilde{44}, \kappa\tilde{43}, \kappa\tilde{42}, \kappa\tilde{41}, \kappa\tilde{40}, \kappa\tilde{39}, \kappa\tilde{38}, \kappa\tilde{37}, \kappa\tilde{36}, \kappa\tilde{35}, \kappa 34, \kappa 33, \kappa 32,$
19	$\kappa\tilde{87}, \kappa\tilde{86}, \kappa\tilde{85}, \kappa\tilde{84}, \kappa\tilde{83}, \kappa\tilde{82}, \kappa\tilde{81}, \kappa\tilde{80}, \kappa\tilde{95}, \kappa\tilde{94}, \kappa\tilde{93}, \kappa\tilde{92}, \kappa\tilde{91}, \kappa\tilde{90}, \kappa\tilde{89}, \kappa\tilde{88},$ $\kappa\tilde{79}, \kappa\tilde{78}, \kappa\tilde{77}, \kappa\tilde{76}, \kappa\tilde{75}, \kappa\tilde{74}, \kappa\tilde{73}, \kappa\tilde{72}, \kappa\tilde{71}, \kappa\tilde{70}, \kappa\tilde{69}, \kappa\tilde{68}, \kappa\tilde{67}, \kappa\tilde{66}, \kappa\tilde{65}, \kappa\tilde{64},$
20	$\kappa\tilde{119}, \kappa\tilde{118}, \kappa\tilde{117}, \kappa\tilde{116}, \kappa\tilde{115}, \kappa\tilde{114}, \kappa\tilde{113}, \kappa\tilde{112}, \kappa\tilde{127}, \kappa\tilde{126}, \kappa\tilde{125}, \kappa\tilde{124}, \kappa\tilde{123}, \kappa\tilde{122}, \kappa\tilde{121}, \kappa\tilde{120},$ $\kappa\tilde{111}, \kappa\tilde{110}, \kappa\tilde{109}, \kappa\tilde{108}, \kappa\tilde{107}, \kappa\tilde{106}, \kappa\tilde{105}, \kappa\tilde{104}, \kappa\tilde{103}, \kappa\tilde{102}, \kappa\tilde{101}, \kappa\tilde{100}, \kappa\tilde{99}, \kappa\tilde{98}, \kappa\tilde{97}, \kappa\tilde{96},$

TABLE 7. The 7-round related-key differential characteristic of GIFT-128 with a probability of $2^{-15.83}$.

R	Input Difference	Subkey Difference	Probability
1	0000 0000 0000 0000 00E0 0000 0000 0000	0000 0040 0000 0000 0000 0000 0800 0000	
2	0000 3000 0000 0000 0000 0000 0000 0000	0200 0000 0000 0040 0000 0000 0000 0000	2^{-2}
3	0000 0000 0000 0000 0000 0000 0C00 0000	0000 0000 0200 0000 0000 0040 0000 0000	2^{-5}
4	0000 0060 0000 0000 0000 0000 0000 0000	0000 0000 0000 0000 0200 0000 0000 0040	2^{-7}
5	0000 0000 0000 0000 0000 0000 0000 0000	0000 0400 0000 0000 0000 0000 0200 0000	2^{-9}
6	0000 0020 0000 0000 0000 0000 0000 0000	0080 0000 0000 0400 0000 0000 0000 0000	2^{-9}
7	0000 0000 0000 0000 0400 0400 0200 0000	0000 0000 0080 0000 0000 0400 0000 0000	2^{-11}
8	0000 2220 4000 1100 0000 0000 0000 4440	-	$2^{-15.83}$

For the sake of clarity, the round key and the key bits that need to be guessed in the 1st, 2nd, 3rd, 17th, 18th, 19th, and 20th rounds will be listed in Table 6 according to the key schedule. The bit position that is marked with "∼" above is the key bit that needs to be guessed in the key recovery attack procedure that is introduced in Table 5.

Attack Procedure

1) Since there are no whitening keys at the beginning of GIFT-64, we can choose 2^n structures in the output of the first round of the PermBit operation. Each structure traverses 48 undetermined difference bits in the ΔAK_{in}^{1} of Table 5, that is, the position that is marked "?". Therefore, we can get 2^{n+95} pairs.
2) After the key guess of the first round, we encrypt the pairs by using two different keys, and then only choose the pairs that satisfy the difference of ΔLP_{in}^{2}. That is, the difference of position "1" is 1, the difference of position "0" is 0, and the difference of position "?" is not limited. This step performs a filter with a

TABLE 8. The 10-round related-key differential characteristic of GIFT-128 with a probability of $2^{-72.66}$.

R	Input Difference	Subkey Difference	Probability
1	0000 A000 0CD0 0000 001D 000A 1E00 0000	1000 0000 0000 0000 0000 0000 0000 0000	
2	0000 0000 0000 C000 01C0 0000 0000 01C0	0000 0000 1000 0000 0000 0000 0000 0000	2^{-19}
3	0000 0000 0000 0000 0000 C00C 0000 0000	0000 0000 0000 0000 1000 0000 0000 0000	2^{-31}
4	0000 0000 0000 0600 0000 0000 0000 0000	0000 0000 0000 0000 0000 0000 1000 0000	2^{-35}
5	0000 0000 0000 0000 0000 0000 0000 0000	0400 0000 0000 0000 0000 0000 0000 0000	2^{-37}
6	0000 0000 0000 0000 0000 0000 0000 0000	0000 0000 0400 0000 0000 0000 0000 0000	2^{-37}
7	0000 0400 0000 0000 0000 0000 0000 0000	0000 0000 0000 0000 0400 0000 0000 0000	2^{-37}
8	0000 0000 0100 0000 0000 0000 0400 0000	0000 0000 0000 0000 0000 0000 0400 0000	2^{-39}
9	0020 0200 0000 0010 0000 0000 0040 0040	0100 0000 0000 0000 0000 0000 0000 0000	2^{-44}
10	0200 0011 0008 0000 4000 0044 2400 0000	0000 0000 0100 0000 0000 0000 0000 0000	2^{-55}
11	2800 4164 0800 2430 0020 1600 4010 0340	-	$2^{-72.66}$

probability of 2^{-33}, and then the number of expected remaining pairs is 2^{n+62}.

3) Encrypt the remaining pairs by using the key guess of the second round and only choose the pairs that satisfy the difference of ΔLP_{in}^3. This provides a filtering probability of 2^{-20}, and then there are approximately 2^{n+42} pairs left.
4) After encrypting the remaining pairs for 20 rounds, guess the 20th and 19th rounds' keys to decrypt the corresponding ciphertexts and leave only the pair that satisfies the difference of ΔAK_{out}^{18}. This step produces a filter with a probability of 2^{-12}, and then the number of expected remaining pairs is 2^{n+30}.
5) After the key guess of the 18th round, decrypt the ciphertexts corresponding to the remaining pairs and only choose the pairs that satisfy the difference of ΔAK_{out}^{17}. This step provides a filter with a probability of 2^{-41}, and then the number of expected remaining pairs is 2^{n-11}.
6) Decrypt the ciphertexts corresponding to the remaining pairs by using the key guess of the 17th round and only

TABLE 9. The 12-round related-key differential characteristic of GIFT-64 with a probability of 2^{-37}.

R	Input Difference	Subkey Difference	Probability
1	0000 0000 0000 0000	0100 4100 0000 0002 0000 0000 0000 0000	
2	0000 0000 0000 0000	0000 0000 0100 4100 0000 0002 0000 0000	1
3	0000 0000 0000 0000	0000 0000 0000 0000 0100 4100 0000 0002	1
4	0000 0000 0000 0010	0000 0020 0000 0000 0000 0000 0100 4100	1
5	0101 000B 0000 0000	0040 1004 0000 0020 0000 0000 0000 0000	2^{-3}
6	A000 0000 0000 4100	0000 0000 0040 1004 0000 0020 0000 0000	2^{-12}
7	0000 0000 1009 000C	0000 0000 0000 0000 0040 1004 0000 0020	2^{-20}
8	0000 0004 0000 0090	0000 0200 0000 0000 0000 0000 0040 1004	2^{-28}
9	0000 0000 0000 0000	0010 0041 0000 0200 0000 0000 0000 0000	2^{-34}
10	0000 0000 0000 0000	0000 0000 0010 0041 0000 0200 0000 0000	2^{-34}
11	0000 0000 0000 0000	0000 0000 0000 0000 0010 0041 0000 0200	2^{-34}
12	0000 0010 0000 0000	0000 2000 0000 0000 0000 0000 0010 0041	2^{-34}
13	0000 0800 0102 0001	-	2^{-37}

TABLE 10. The 19-round attack on GIFT-64 that is constructed with the 12-round related-key differential characteristic.

State	Difference
$P = \Delta SB_{in}^1$	???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ????
ΔLP_{in}^1	???0 0??? ?0?? ??0? ???0 0??? ?0?? ??0? ???0 0??? ?0?? ??0? ?100 0?00 10?0 110?
ΔAK_{in}^1	???? ???? ???? 1100 ???? ???? ???? 1100 0000 0000 0000 0000 ???? ???? ???? ????
ΔSB_{in}^2	???? ???? ???? 1100 ???? ???? ???? 1100 0000 0000 0000 0000 ???? ???? ???? ????
ΔLP_{in}^2	00?0 000? ?001 0100 00?0 000? ?000 0100 0000 0000 0000 0000 000? 1000 0?00 00?0
ΔAK_{in}^2	0001 0000 0000 0000 ?1?? ?1?? 0000 0000 0000 0000 0000 1??? 0000 0000 0000 0000
ΔSB_{in}^3	0000 0000 0000 0000 ?1?? ?1?? 0000 0000 0000 0000 0000 1??? 0000 0000 0000 0000
ΔLP_{in}^3	0000 0000 0000 0000 0010 0010 0000 0000 0000 0000 0000 0001 0000 0000 0000 0000
ΔAK_{in}^3	0000 0000 0000 0000 0000 0011 0000 0000 0000 0000 0000 0000 0000 0000 0001 0000
ΔSB_{in}^4	0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
...	...
ΔAK_{out}^{15}	0000 0000 0000 0000 0000 1000 0000 0000 0000 0001 0000 0010 0000 0000 0000 0001
ΔSB_{out}^{16}	0000 0000 0000 0000 0000 ??11 0000 0000 0000 ???? 0000 ???? 0000 0000 0000 ????
ΔLP_{out}^{16}	0000 0010 00?0 ?000 0000 0001 0?0? 0?00 0000 ?000 ?0?0 00?0 ?000 0?00 0?0? 000?
ΔAK_{out}^{16}	0000 0010 00?0 ?000 0000 0001 0?0? 0?00 0000 ?000 ?0?0 00?0 ?000 0?00 0?0? 000?
ΔSB_{out}^{17}	0000 ???? ???? ???? 0000 ???? ???? ???? 0000 ???? ???? ???? ???? ???? ???? ????
ΔLP_{out}^{17}	?0?? ?0?? ?0?? ???? ??0? ??0? ??0? ???? ???0 ???0 ???0 ???? 0??? 0??? 0??? ????
ΔAK_{out}^{17}	?0?? ?0?? ?0?? ???? ??0? ??0? ??0? ???? ???0 ???0 ???0 ???? 0??? 0??? 0??? ????
ΔSB_{out}^{18}	???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ????
ΔLP_{out}^{18}	???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ????
ΔAK_{out}^{18}	???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ????
ΔSB_{in}^{19}	???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ????
ΔLP_{in}^{19}	???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ????
$C = \Delta AK_{in}^{19}$	???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ???? ????

choose the pairs that satisfy the difference of ΔAK_{out}^{16}. This step provides a filter with a probability of 2^{-12}. Then, the number of expected remaining pairs is 2^{n-23}. Thus, there are 2^{n-23} pairs that satisfy the input and output differential of the 13-round distinguisher and will be left for the random key.

7) For a pair that obeys the difference of ΔAK_{in}^{1}, the average probability of satisfying the input of the distinguisher that is shown in Table 4 is 2^{-53}. A pair that is encrypted with a wrong key will meet the output differential of the 16th round with a probability of 2^{-64}. Additionally, the pairs that are encrypted with the right key will meet it with a probability of $2^{-47.83}$, so $2^{n+95} \times 2^{-53} \times 2^{-47.83} = 2^{n-5.83}$ pairs should be left for a right key. Here, we choose $n = 8$, and the data complexity is $2^{8+48} = 2^{56}$.

The 19-round key recovery attack on GIFT-64 is similar to that of the 20-round, therefore, here we skip the details and just list the 19-round related-key differential characteristic in Table 10 in Appendix A, which is constructed using the differential characteristic in Table 9 as the distinguisher. During this analysis, each of the 2^n structures traverses 44 nonzero bits in ΔAK_{in}^{1} of Table 10, resulting in a total of 2^{n+87} pairs. There are $2^{n+87} \times 2^{-51} \times 2^{-37} = 2^{n-1}$ pairs that will fulfil the differential trail with the right key guess. We choose $n = 3$, so about 4 pairs should be left for a right key. The data complexity is $2^{3+44} = 2^{47}$.

V. CONCLUSION

In this paper, we perform related-key differential cryptanalysis on the GIFT algorithm and give several differential characteristics with higher probabilities. Then, the 19-round and 20-round GIFT-64 key recovery attacks with respective data complexities of 2^{47} and 2^{56} are carried out based on those characteristics. Our research results are progressively compared to the previous ones, and we hope that these results can be used for better analysis in the future.

APPENDIX

See Tables 7–10.

REFERENCES

[1] Z. Zhou, B. Wang, Y. Guo, and Y. Zhang, "Blockchain and computational intelligence inspired incentive-compatible demand response in Internet of electric vehicles," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 3, no. 3, pp. 205–216, May 2019, doi: 10.1109/TETCI.2018.2880693.

[2] Z. Zhou, B. Wang, M. Dong, and K. Ota, "Secure and efficient vehicle-to-grid energy trading in cyber physical systems: Integration of blockchain and edge computing," *IEEE Trans. Syst., Man, Cybern. Syst.*, to be published, doi: 10.1109/TSMC.2019.2896323.

[3] D. Sehrawat, N. S. Gill, and M. Devi, "Comparative analysis of lightweight block ciphers in IoT-enabled smart environment," in *Proc. 6th Int. Conf. Signal Process. Integr. Netw. (SPIN)*, Mar. 2019, pp. 915–920, doi: 10.1109/SPIN.2019.8711697.

[4] A. Bogdanov, L. R. Knudsen, G. Leander, C. Paar, A. Poschmann, M. J. B. Robshaw, Y. Seurin, and C. Vikkelsoe, "PRESENT: An ultra-lightweight block cipher," in *Proc. 9th Int. Workshop Cryptograph. Hardw. Embedded Syst. (CHES)*, in Lecture Notes in Computer Science, Vienna, Austria, vol. 4727, P. Paillier and I. Verbauwhede, Eds. Berlin, Germany: Springer, Sep. 2007, pp. 450–466, doi: 10.1007/978-3-540-74735-2_31.

[5] Z. Xiang, W. Zhang, Z. Bao, and D. Lin, "Applying MILP method to searching integral distinguishers based on division property for 6 lightweight block ciphers," in *Proc. 22nd Int. Conf. Theory Appl. Cryptol. Inf. Secur.*, in Lecture Notes in Computer Science, Hanoi, Vietnam, vol. 10031, J. H. Cheon and T. Takagi, Eds. Berlin, Germany: Springer, Dec. 2016, pp. 648–678, doi: 10.1007/978-3-662-53887-6_24.

[6] S. Sun, L. Hu, P. Wang, K. Qiao, X. Ma, and L. Song, "Automatic security evaluation and (related-key) differential characteristic search: Application to SIMON, PRESENT, LBlock, DES(L) and other bit-oriented block ciphers," in *Proc. 20th Int. Conf. Theory Appl. Cryptol. Inf. Secur.*, in Lecture Notes in Computer Science, Kaoshiung, Taiwan, vol. 8873, P. Sarkar and T. Iwata, Eds. Berlin, Germany: Springer, Dec. 2014, pp. 158–178, doi: 10.1007/978-3-662-45611-8_9.

[7] S. Banik, S. K. Pandey, T. Peyrin, S. M. Sim, Y. Todo, and Y. Sasaki, "GIFT: A small PRESENT," IACR Cryptol. ePrint Arch., Tech. Rep. 2017/622, 2017. [Online]. Available: http://eprint.iacr.org/2017/622

[8] E. Biham and A. Shamir, "Differential cryptanalysis of DES-like cryptosystems," *J. Cryptol.*, vol. 4, no. 1, pp. 3–72, 1991, doi: 10.1007/BF00630563.

[9] E. Biham and A. Shamir, "Differential cryptanalysis of Snefru, Khafre, REDOC-II, LOKI and Lucifer," in *Proc. 11th Annu. Int. Cryptol. Conf.*, in Lecture Notes in Computer Science, Santa Barbara, CA, USA, vol. 576, J. Feigenbaum, Ed. Springer, Aug. 1991, pp. 156–171, doi: 10.1007/3-540-46766-1_11.

[10] E. Biham, "New types of cryptanalytic attacks using related keys," *J. Cryptol.*, vol. 7, no. 4, pp. 229–246, 1994, doi: 10.1007/BF00203965.

[11] J. Kelsey, B. Schneier, and D. A. Wagner, "Key-schedule cryptanalysis of IDEA, G-DES, GOST, SAFER, and triple-DES," in *Proc. 16th Annu. Int. Cryptol. Conf.*, in Lecture Notes in Computer Science, Santa Barbara, CA, USA, vol. 1109, N. Koblitz, Ed. Springer, Aug. 1996, pp. 237–251, doi: 10.1007/3-540-68697-5_19.

[12] N. Mouha, Q. Wang, D. Gu, and B. Preneel, "Differential and linear cryptanalysis using mixed-integer linear programming," in *Proc. 7th Int. Conf. Inf. Secur. Cryptol. (Inscrypt)*, in Lecture Notes in Computer Science, Beijing, China, vol. 7537, C. Wu, M. Yung, and D. Lin, Eds. Berlin, Germany: Springer, Dec. 2011, pp. 57–76, doi: 10.1007/978-3-642-34704-7_5.

[13] W. Zhang and V. Rijmen, "Division cryptanalysis of block ciphers with a binary diffusion layer," *IET Inf. Secur.*, vol. 13, no. 2, pp. 87–95, 2019, doi: 10.1049/iet-ifs.2018.5151.

[14] Y. Hao, L. Jiao, C. Li, W. Meier, Y. Todo, and Q. Wang, "Observations on the dynamic cube attack of 855-round TRIVIUM from crypto'18," IACR Cryptol. ePrint Arch., Tech. Rep. 2018/972, 2018. [Online]. Available: https://eprint.iacr.org/2018/972

[15] Y. Hao, Y. Todo, C. Li, T. Isobe, W. Meier, and Q. Wang, "Improved division property based cube attacks exploiting algebraic properties of superpoly," in *Proc. 38th Annu. Int. Cryptol. Conf.*, in Lecture Notes in Computer Science, Santa Barbara, CA, USA, vol. 10991, H. Shacham and A. Boldyreva, Eds. Cham, Switzerland: Springer, Aug. 2018, pp. 275–305, doi: 10.1007/978-3-319-96884-1_10.

[16] S. Huang, X. Wang, G. Xu, M. Wang, and J. Zhao, "Conditional cube attack on reduced-round keccak sponge function," in *Proc. 36th Annu. Int. Conf. Theory Appl. Cryptograph. Techn.*, in Lecture Notes in Computer Science, Paris, France, vol. 10211, J. Coron and J. B. Nielsen, Eds. Cham, Switzerland: Springer, May 2017, pp. 259–288, doi: 10.1007/978-3-319-56614-6_9.

[17] L. Song, J. Guo, D. Shi, and S. Ling, "New MILP modeling: Improved conditional cube attacks on keccak-based constructions," in *Proc. 24th Int. Conf. Theory Appl. Cryptol. Inf. Secur.*, in Lecture Notes in Computer Science, Brisbane, QLD, Australia, vol. 11273, T. Peyrin and S. D. Galbraith, Eds. Cham, Switzerland: Springer, Dec. 2018, pp. 65–95, doi: 10.1007/978-3-030-03329-3_3.

[18] Z. Li, W. Bi, X. Dong, and X. Wang, "Improved conditional cube attacks on Keccak keyed modes with MILP method," in *Proc. 23rd Int. Conf. Theory Appl. Cryptol. Inf. Secur.*, in Lecture Notes in Computer Science, Hong Kong, vol. 10624, T. Takagi and T. Peyrin, Eds. Cham, Switzerland: Springer, Dec. 2017, pp. 99–127, doi: 10.1007/978-3-319-70694-8_4.

[19] B. Zhu, X. Dong, and H. Yu, "MILP-based differential attack on round-reduced GIFT," in *Proc. Cryptograph. Track RSA Conf. Topics Cryptol. (CT-RSA)*, in Lecture Notes in Computer Science, San Francisco, CA, USA, vol. 11405, M. Matsui, Ed. Springer, Mar. 2019, pp. 372–390, doi: 10.1007/978-3-030-12612-4_19.

[20] Y. Sasaki, "Integer linear programming for three-subset meet-in-the-middle attacks: Application to GIFT," in *Proc. 13th Int. Workshop Secur. Adv. Inf. Comput. Secur. (IWSEC)*, Sendai, Japan, in Lecture Notes in Computer Science, vol. 11049, A. Inomata and K. Yasuda, Eds. Cham, Switzerland: Springer, Sep. 2018, pp. 227–243, doi: 10.1007/978-3-319-97916-8_15.

[21] Y. Liu and Y. Sasaki, "Related-key boomerang attacks on GIFT with automated trail search including BCT effect," in *Proc. 24th Australas. Conf. Inf. Secur. Privacy (ACISP)*, Christchurch, New Zealand, in Lecture Notes in Computer Science, vol. 11547, J. Jang-Jaccard and F. Guo, Eds. Cham, Switzerland: Springer, Jul. 2019, pp. 555–572, doi: 10.1007/978-3-030-21548-4_30.

[22] S. Patranabis, N. Datta, D. Jap, J. Breier, S. Bhasin, and D. Mukhopadhyay, "SCADFA: Combined SCA+DFA attacks on block ciphers with practical validations," *IEEE Trans. Comput.*, vol. 68, no. 10, pp. 1498–1510, Apr. 2019, doi: 10.1109/TC.2019.2913644.

[23] S. Sun, L. Hu, M. Wang, P. Wang, K. Qiao, X. Ma, D. Shi, L. Song, and K. Fu, "Towards finding the best characteristics of some bit-oriented block ciphers and automatic enumeration of (related-key) differential and linear characteristics with predefined properties," Cryptol. ePrint Arch., Rep. 747/2014, 2014.

[24] (2007). *An Open Source Mathematical Computing Platform*. [Online]. Available: http://www.sagemath.org/

[25] Y. Sasaki and Y. Todo, "New algorithm for modeling S-box in MILP based differential and division trail search," in *10th Int. Conf. Innov. Secur. Solutions Inf. Technol. Commun. (SecITC)*, Bucharest, Romania, in Lecture Notes in Computer Science, vol. 10543, P. Farshim and E. Simion, Eds. Cham, Switzerland: Springer, Jun. 2017, pp. 150–165, doi: 10.1007/978-3-319-69284-5_11.

[26] P. Zhang and W. Zhang, "Differential cryptanalysis on block cipher skinny with MILP program," *Secur. Commun. Netw.*, vol. 2018, pp. 3780407-1–3780407-11, 2018, doi: 10.1155/2018/3780407.

MEICHUN CAO is currently pursuing the Ph.D. degree with the School of Information Science and Engineering, Shandong Normal University, China. Her current research interests include information security, and analysis and design of block ciphers.

WENYING ZHANG received the Ph.D. degree in cryptography from the Department of Information Research, PLA Information Engineering University, Zhengzhou, China, in June 2004. From July 2004 to September 2006, she was a Postdoctoral Fellow with the Institute of Software, Chinese Academy of Sciences, Beijing, China. She is currently a Professor with the School of Information Science and Engineering, Shandong Normal University, China. Her research interests include cryptography, Boolean function, and hash function analysis.