

# Improved Attacks on GIFT-64

Ling Sun<sup>1,2,3</sup>, Wei Wang<sup>1,3</sup>, and Meiqin Wang(✉)<sup>1,3</sup>

<sup>1</sup> Key Laboratory of Cryptologic Technology and Information Security,  
Ministry of Education, Shandong University, Jinan, China

<sup>2</sup> State Key Laboratory of Cryptology, P. O. Box 5159, Beijing, 100878, China

<sup>3</sup> School of Cyber Science and Technology, Shandong University, Qingdao, China  
{lingsun, weiwangsdu, mqwang}@sdu.edu.cn

**Abstract.** One of the well-known superiorities of GIFT-64 over PRESENT lies in the correction of the strong linear hull effect. However, apart from the investigation of the 9-round linear hull effect in the design document, we find no linear attack result on GIFT-64. Although we do not doubt the security of GIFT-64 regarding the linear cryptanalysis, the actual resistance of the cipher to the linear attack should be evaluated since it promotes a comprehensive perception of the soundness of GIFT-64. Motivated by this observation, we implement an automatic search and find a 12-round linear distinguisher whose dominating trail is an optimal linear characteristic. Following that, the first 19-round linear attack is launched by utilising the newly identified distinguisher. On the other side, we notice that the previous differential attack of GIFT-64 covering 20 rounds claims the entire codebook. To reduce the data complexity of the 20-round attack, we apply the automatic method to exhaustively check 13-round differential trails with probabilities no less than  $2^{-64}$  and identify multiple 13-round differentials facilitating 20-round attacks without using the full codebook. One of the candidate differentials with the maximum probability and the minimum number of guessed subkey bits is then employed to realise the first 20-round differential attack without relying on the complete codebook. Given the newly obtained results, we conjecture that the resistances of GIFT-64 against differential and linear attacks do not have a significant gap. Also, we note that the attack results in this paper are far from threatening the security of GIFT-64.

**Keywords:** Linear cryptanalysis · Differential cryptanalysis · GIFT-64.

## 1 Introduction

GIFT [4] is a lightweight block cipher motivated by the PRESENT [10] design strategy. The comprehensive treatment on the linear layer and the S-box makes it one of the most energy-efficient ciphers as of today. It outperforms even SIMON [5] and SKINNY [6] for round-based implementations. Another bonus of the wise organisation is the significantly reduced linear hull effect [14], which constitutes the weak point of PRESENT.

Because of the good performance in hardware and software implementations, GIFT acted as the underlying primitives of many lightweight designs, such

as GIFT-COFB [3], HyENA [12], LOTUS-AEAD and LOCUS-AEAD [11], and SUNDABE-GIFT [2]. Notably, GIFT-COFB has been selected as one of the ten finalists of the ongoing NIST Lightweight Cryptography standardisation project<sup>4</sup>. Thus, evaluating the security level of GIFT is essential.

When we investigate the security of GIFT-64, which is one version of GIFT with the 64-bit block size, an interesting phenomenon is identified. Many results are focusing on the differential attack [7] of GIFT-64. As in Table 1, Chen et al. [13] proposed 20-round and 21-round differential attacks with the full codebook. To reduce the data requirement, they utilised multiple differentials and provided a 20-round attack without using the full codebook. However, few works consider the security of the cipher regarding the linear attack [21]. To be precise, apart from the study of the 9-round linear hull effect in the design document [4], we find no linear attack result on GIFT-64. Although we do not doubt the security of GIFT-64 regarding the linear cryptanalysis, the actual resistance of the cipher to the linear attack should be evaluated since it promotes a comprehensive perception of the soundness of GIFT-64. This observation drives the work in this paper.

## 1.1 Contributions

This paper focuses on the security of GIFT-64 regarding the linear and differential attacks. The search of distinguishers is accomplished with the automatic method in [29], which is realised via the Boolean Satisfiability Problem (SAT). The contributions are fourfold.

*Estimations for expected linear potentials of all 12-round linear approximations with dominating trails being the optimal ones.* After discovering 5120 optimal 12-round linear trails with the SAT solver, we perceive that all of them can launch valid 19-round linear attacks. However, in order to identify relatively good distinguishers among these candidates, we turn attention to the expected linear potentials of the 5120 linear approximations containing the 5120 trails. We exhaustively search for all trails belonging to the linear approximations with correlations larger than  $2^{-40}$  and generate rough estimations for the expected linear potentials of the 5120 linear approximations. The estimations facilitate the selection of linear distinguisher in the attack.

*The first 19-round linear attack result on GIFT-64.* Among the 5120 candidate distinguishers, we choose one linear approximation maintaining the minimum number of guessed subkey bits in the subkey enumeration phase. This linear approximation is then exploited to drive a 19-round attack. The data complexity is  $2^{63.26}$  known plaintexts, the time complexity is  $2^{127.00}$  19-round of encryptions, and the memory complexity is about  $2^{60.00}$ . As far as we know, this is the first linear attack result on GIFT-64.

---

<sup>4</sup> <https://csrc.nist.gov/projects/lightweight-cryptography>

*Estimations for probabilities of 2392 differentials facilitating 20-round differential attacks.* We check all 13-round differential trails with probabilities more significant than  $2^{-64}$  and notice that none can perform valid 20-round differential attacks for the considerable time complexities. Thus, we manage to study trails with probabilities being  $2^{-64}$ . With the SAT solver, we discover 92768 trails with probabilities being  $2^{-64}$  and identify 2392 differentials having the possibility to actualise valid 20-round differential attacks. Then, in order to get approximate evaluations for the probabilities of the 2392 differentials, we apply the SAT solver to search for all differential trails within each differential with probabilities being larger than  $2^{-71}$ . The experimental results guide the decision of the distinguisher in the differential attack.

*The first 20-round differential attack without using the full codebook.* We notice that the previous 20-round differential attack in [13] demanded the full codebook. To reduce the data complexity of the 20-round attack, we detect a new 13-round distinguisher with the maximum differential probability and the minimum number of guessed subkey bits. Based on this distinguisher, we realise the first 20-round differential attack without relying on the entire codebook. The data complexity is  $2^{62.58}$  chosen plaintexts, the time complexity is  $2^{125.50}$  19-round of encryptions, and the memory complexity is about  $2^{62.58}$ . A summary of cryptanalytic results on GIFT-64 to date can be found in Table 1.

**Table 1.** Summary of cryptanalytic results on GIFT-64.

Round	Method	Setting	Time	Data	Memory	Ref.
14	Integral	SK	$2^{97.00}$	$2^{63.00}$	-	[4]
15	MITM	SK	$2^{120.00}$	$2^{64.00}$	-	[4]
15	MITM	SK	$2^{112.00}$	-	-	[23]
19	Differential	SK	$2^{112.0}$	$2^{63.00}$	-	[31]
<b>19</b>	<b>Linear</b>	<b>SK</b>	<b><math>2^{127.11}</math></b> Ⓢ	<b><math>2^{62.96}</math></b>	<b><math>2^{60.00}</math></b>	<b>Sect. 3</b>
<b>20</b>	<b>Differential</b>	<b>SK</b>	<b><math>2^{125.50}</math></b>	<b><math>2^{62.58}</math></b>	<b><math>2^{62.58}</math></b>	<b>Sect. 4</b>
20	Differential	SK	$2^{101.68}$ *	$2^{64.00}$	$2^{96.00}$	[13]
21	Differential	SK	$2^{107.61}$ *	$2^{64.00}$	$2^{96.00}$	[13]
20	Multiple differential	SK	$2^{112.68}$ *	$2^{62.00}$	$2^{112.00}$	[13]
23	Boomerang	RK	$2^{126.60}$	$2^{63.30}$	-	[19]
24	Rectangle	RK	$2^{106.00}$	$2^{63.78}$	$2^{64.10}$	[16]
25	Rectangle	RK	$2^{120.92}$	$2^{63.78}$	$2^{64.10}$	[16]
26	Differential	RK	$2^{123.23}$	$2^{60.96}$	$2^{102.86}$	[30]

\* Attacks in [13] only computed the time complexity in the subkey enumeration phase.

Ⓢ The success probability of the attack is 60%.

*Outline of the paper.* In Sect. 2, the structure of **GIFT-64**, the utilised automatic approach, and the methods to compute the complexities are recalled. Sect. 3 presents the first 19-round linear attack on the cipher. The first 20-round differential attack on **GIFT-64** without using the full codebook is proposed in Sect. 4. Sect. 5 concludes the paper.

## 2 Preliminaries

In this section, we first review the objective primitive of this work. Then, the automatic method utilised to search for differential and linear distinguishers is briefly introduced. At last, the methods to evaluate the complexities in the differential and linear attacks are recalled.

### 2.1 Description of GIFT-64

**GIFT** [4] is a family of lightweight block ciphers composed of two versions. The version with the 64-bit block size is denoted as **GIFT-64** in this paper. **GIFT-64** is a 28-round Substitution-Permutation Network (SPN) cipher and has a key length of 128-bit. The plaintext is presented as  $b_0b_1 \cdots b_{63}$ , where  $b_0$  stands for the most significant bit. We use  $K = k_0 \| k_1 \| \cdots \| k_7$  to represent the 128-bit key, where  $k_i$ 's are 16-bit words. Each round of **GIFT-64** consists of three steps: **SubCells**, **PermBits**, and **AddRoundKey**.

**SubCells** An invertible 4-bit S-box  $GS$ , provided in the following, is applied to every nibble of the cipher state.

$x$	0x0	0x1	0x2	0x3	0x4	0x5	0x6	0x7	0x8	0x9	0xa	0xb	0xc	0xd	0xe	0xf
$GS(x)$	0x1	0xa	0x4	0xc	0x6	0xf	0x3	0x9	0x2	0xd	0xb	0x7	0x5	0x0	0x8	0xe

**PermBits** The bit permutation operation maps the  $i$ -th bit of the cipher state to the  $P(i)$ -th bit, i.e.,  $b_{P(i)} \leftarrow b_i$ ,  $i \in \{0, 1, \dots, 63\}$ . The specification of  $P$  is given as follows.

$i$	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$P(i)$	48	1	18	35	32	49	2	19	16	33	50	3	0	17	34	51
$i$	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
$P(i)$	52	5	22	39	36	53	6	23	20	37	54	7	4	21	38	55
$i$	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
$P(i)$	56	9	26	43	40	57	10	27	24	41	58	11	8	25	42	59
$i$	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
$P(i)$	60	13	30	47	44	61	14	31	28	45	62	15	12	29	46	63

**AddRoundKey** This step includes adding the round key and the round constant. Since adding the round constant does not alter validities of attacks in this paper, we do not introduce it.

As for the adding round key operation, after extracting a 32-bit round key  $RK$  from the key state, we split it into two 16-bit words as  $RK = U\|V = u_0u_1 \cdots u_{15}\|v_0v_1 \cdots v_{15}$ . Then,  $U$  and  $V$  are XORed with the cipher state as  $b_{4 \cdot i+2} \leftarrow b_{4 \cdot i+2} \oplus u_i$ ,  $b_{4 \cdot i+3} \leftarrow b_{4 \cdot i+3} \oplus v_i$ ,  $i \in \{0, 1, \dots, 15\}$ .

The key schedule of GIFT-64 is carefully created so that the hardware and software implementations of the cipher are optimised.

**Key schedule** Before updating the key state, the round key is first extracted from the key state as  $RK = U\|V = k_6\|k_7$ . Then, the key state is updated as  $k_0\|k_1\| \cdots \|k_7 \leftarrow (k_6 \ggg 2)\|(k_7 \ggg 12)\|k_0\| \cdots \|k_4\|k_5$ .

Note that we only recall the necessary message about the cipher. For more details, please refer to [4].

## 2.2 Searching for Differential and Linear Distinguishers of GIFT-64

The cornerstones in differential and linear attacks are distinguishers exhibiting non-random cryptanalytic features. In this work, we exploit the automatic method in [29] to accomplish the search of differential and linear distinguishers. The underlying mathematical problem that facilitates the automatic search is the Boolean Satisfiability Problem (SAT), which studies the satisfiability of a given Boolean formula. A SAT problem is said satisfiable if there exists an assignment of Boolean values to variables so that the formula is evaluated to be **True**. Although the SAT problem is proved to be NP-complete [15], modern SAT solvers can handle practical problems with millions of variables.

Almost all existing SAT solvers accept Boolean formulas in Conjunctive Normal Form (CNF) as inputs. That is, the Boolean formula in question should be expressed as a conjunction ( $\wedge$ ) of one or more clauses, where a clause is a disjunction ( $\vee$ ) of (possibly negated) Boolean variables. Thus, in cryptanalysis, the automatic search is realised by converting the distinguisher searching problem into SAT problems in CNF.

In the following, we take the search of differential distinguisher for GIFT-64 as an illustration and remind readers that the distinguisher searching in the linear setting can be implemented likewise.

We start with the search for differential trails. According to the functionality, the Boolean expressions in the SAT problem can be partitioned into two groups. The first group is used to track the differential propagation, and the second one characterises the differential probability of the trail.

*Group 1: Propagating differences inside the cipher.* As the PermBits operation only alters the positions of bits in the cipher state, depicting the differential propagation across the cipher comes down to the description of the S-box  $GS$ . Let  $\mathbf{x} \in \mathbb{F}_2^4$  and  $\mathbf{y} \in \mathbb{F}_2^4$  be the input and output differences of  $GS$ , respectively. The entries in the differential distribution table (DDT) of  $GS$  have five possible evaluations, which are 0, 2, 4, 6, and 16. Correspondingly, the range of differential probabilities is  $\{0, 2^{-3}, 2^{-2}, 2^{-1.415}, 1\}$ . As in [29], for each S-box, four more Boolean variables  $\rho_0, \rho_1, \rho_2$ , and  $\varepsilon$  are introduced to encode the information

about the probability. For a differential propagation with nonzero probability  $p$ , the value of  $\rho_0 \|\rho_1 \|\rho_2 \|\varepsilon$  meets the following rule

$$\rho_0 \|\rho_1 \|\rho_2 \|\varepsilon = \begin{cases} 1110, & \text{if } p = 2^{-3} \\ 0110, & \text{if } p = 2^{-2} \\ 0011, & \text{if } p = 2^{-1.415} \\ 0000, & \text{if } p = 1 \end{cases}.$$

Note that the opposite number of the binary logarithm of  $p$  equals  $\rho_0 + \rho_1 + \rho_2 + 0.415 \cdot \varepsilon$ . Next, we define a 12-bit Boolean function  $f(\mathbf{x} \|\mathbf{y} \|\rho_0 \|\rho_1 \|\rho_2 \|\varepsilon)$  as

$$f(\mathbf{x} \|\mathbf{y} \|\rho_0 \|\rho_1 \|\rho_2 \|\varepsilon) = \begin{cases} 1, & \begin{array}{l} \text{if } \mathbf{x} \rightarrow \mathbf{y} \text{ is a possible propagation} \\ \text{with } -\log_2(p) = \sum_{i=0}^2 \rho_i + 0.415 \cdot \varepsilon \end{array} \\ 0, & \text{otherwise} \end{cases}$$

After simplifying the expression of  $f$  with the off-the-shelf software Logic Friday [22], we obtain a set of Boolean equations that draws the relation among  $\mathbf{x}$ ,  $\mathbf{y}$ ,  $\rho_0$ ,  $\rho_1$ ,  $\rho_2$ , and  $\varepsilon$ . Please refer to [29] for more details.

*Group 2: Monitoring the differential probability of trail.* Suppose that we aim at  $r$ -round trails. Denote the auxiliary variables for the  $j$ -th S-box in the  $i$ -th round as  $\rho_k^{(i,j)}$  and  $\varepsilon^{(i,j)}$ , where  $0 \leq i \leq r-1$ ,  $0 \leq j \leq 15$ , and  $0 \leq k \leq 2$ . The weight, which equals the opposite number of the binary logarithm of the probability,

of the differential trail should be  $\sum_{i=0}^{r-1} \sum_{j=0}^{15} \sum_{k=0}^2 \rho_k^{(i,j)} + 0.415 \cdot \left( \sum_{i=0}^{r-1} \sum_{j=0}^{15} \varepsilon^{(i,j)} \right)$ , and

we call the first and second terms in this formula the *integral* and *decimal parts* of the differential probability, respectively. In theory, given a prospective value  $\omega \in \mathbb{R}^5$  for the weight of the trail, the automatic search should fulfil the search of trails with

$$\sum_{i=0}^{r-1} \sum_{j=0}^{15} \sum_{k=0}^2 \rho_k^{(i,j)} + 0.415 \cdot \left( \sum_{i=0}^{r-1} \sum_{j=0}^{15} \varepsilon^{(i,j)} \right) \leq \omega. \quad (1)$$

However, as the SAT problem is oriented to binary variables, we do not find a feasible method to interpret decimal arithmetics with Boolean expressions. Thus, we transform the original restriction in Eq. (1) into two inequalities oriented to integers. To be specific, the predicted value  $\omega$  is expressed as  $\omega = \omega_{\text{I}} + 0.415 \cdot \omega_{\text{D}}$  with  $\omega_{\text{I}}$  and  $\omega_{\text{D}}$  being two non-negative integers. Accordingly, the objective function of the SAT problem consists of the following two inequalities

$$\sum_{i=0}^{r-1} \sum_{j=0}^{15} \sum_{k=0}^2 \rho_k^{(i,j)} \leq \omega_{\text{I}} \quad \text{and} \quad \sum_{i=0}^{r-1} \sum_{j=0}^{15} \varepsilon^{(i,j)} \leq \omega_{\text{D}}.$$

---

<sup>5</sup>  $\mathbb{R}$  stands for the rational number field.

Note that these two restrictions are cardinality constraints of the form  $\sum_{i=0}^{n-1} x_i \leq k$ , where  $k$  is a non-negative integer.

◊ If  $k = 0$ , this constraint is equivalent to the following  $n$  Boolean expressions

$$\overline{x_i} = 1, \quad 0 \leq i \leq n - 1.$$

◊ In the case of  $k > 0$ , according to the method in [20], this kind of constraint can be converted into Boolean expressions with the sequential encoding method [25]. Precisely, after introducing  $(n - 1) \cdot k$  auxiliary Boolean variables  $v_{i,j}$  ( $0 \leq i \leq n - 2, 0 \leq j \leq k - 1$ ), the following clauses should be satisfied simultaneously if the relation  $\sum_{i=0}^{n-1} x_i \leq k$  holds

$$\left. \begin{array}{l} \overline{x_0} \vee v_{0,0} = 1 \\ \overline{v_{0,j}} = 1, \quad 1 \leq j \leq k - 1 \\ \overline{x_i} \vee v_{i,0} = 1 \\ \overline{v_{i-1,0}} \vee v_{i,0} = 1 \\ \overline{x_i} \vee \overline{v_{i-1,j-1}} \vee v_{i,j} = 1 \\ \overline{v_{i-1,j}} \vee v_{i,j} = 1 \\ \overline{x_i} \vee \overline{v_{i-1,k-1}} = 1 \\ \overline{x_{n-1}} \vee \overline{v_{n-2,k-1}} = 1 \end{array} \right\} \left. \begin{array}{l} 1 \leq j \leq k - 1 \\ 1 \leq i \leq n - 2 \end{array} \right\}$$

For now, we complete the creation of SAT problems for the search of differential trails with the desired probability, and the search for linear characteristics can be realised similarly. The solver utilised in this work is CryptoMiniSat5 [28].

Lastly, we note that there might be trails with the same input and output differences (resp., masks), and the distinguishers operating in attacks are differentials (resp., linear approximations) comprising all trails sharing the same input and output differences (resp., masks). Thus, after fixing the input and output differences (resp., masks) in the differential (resp., linear) distinguisher, the differential (resp., linear hull) effect is also evaluated by applying the SAT solver to search for more trails within the differential (resp., linear approximation). Please find in [1,17,20,27,29] for more information.

The source codes regarding the search in this paper are publicly available at [https://github.com/SunLing134340/Improved\\_Attacks\\_GIFT64](https://github.com/SunLing134340/Improved_Attacks_GIFT64).

### 2.3 Complexity Analysis of the Differential Attack

Let  $\Delta_{\text{in}} \rightarrow \Delta_{\text{out}}$  be an  $r$ -round differential of an iterated block cipher. According to the Markov cipher theory [18], the probability of a differential is calculated as the sum of probabilities regarding all trails sharing the same input and output differences with the differential. Denote the probability of the  $r$ -round differential

as  $p_0$  and the number of plaintext pairs utilised in the attack as  $N_D$ . Thus, under the right key guess, the counter memorising the number of pairs validating the differential follows a binomial distribution of parameters  $(N_D, p_0)$ . On the other side, suppose that the probability of a pair fulfilling the differential under a wrong key guess is  $p$ . Consequently, the counter follows a binomial distribution of parameters  $(N_D, p)$ . We fix the threshold in the attack as  $\tau_D$ , and the key guess will be accepted if the counter of right pairs is no less than  $\tau_D$ .

The statistical cryptanalysis is always faced with two errors, and we denote by  $\alpha$  the non-detection error probability and  $\beta$  the false alarm error probability. Therefore, the success probability  $P_S$  of the attack equals  $1 - \alpha$ . With the analysis in [8], when  $N_D$  is sufficiently large, the following approximations hold

$$\alpha \approx \frac{(1-p) \cdot \sqrt{\tau_D/N_D}}{(\tau_D/N_D - p) \cdot \sqrt{2 \cdot \pi \cdot N_D \cdot (1 - \tau_D/N_D)}} \cdot \exp \left[ -N_D \cdot D \left( \frac{\tau_D}{N_D} \parallel p \right) \right],$$

$$\beta \approx \frac{p_0 \cdot \sqrt{1 - (\tau_D - 1)/N_D}}{(p_0 - (\tau_D - 1)/N_D) \cdot \sqrt{2 \cdot \pi \cdot (\tau_D - 1)}} \cdot \exp \left[ -N_D \cdot D \left( \frac{\tau_D - 1}{N_D} \parallel p_0 \right) \right],$$

where  $D(p \parallel q) \triangleq p \cdot \ln \left( \frac{p}{q} \right) + (1-p) \cdot \ln \left( \frac{1-p}{1-q} \right)$  is the Kullback-Leibler divergence between two Bernoulli probability distributions with parameters respectively being  $p$  and  $q$ .

## 2.4 Complexity Analysis of the Linear Attack

Denote  $\Gamma_{\text{in}} \rightarrow \Gamma_{\text{out}}$  an  $r$ -round linear approximation of an iterated block cipher with  $n$ -bit block size. Suppose that the absolute value of the correlation regarding the dominating linear characteristic  $\mu = (\mu_0, \mu_1, \dots, \mu_r)$  with  $\mu_0 = \Gamma_{\text{in}}$  and  $\mu_r = \Gamma_{\text{out}}$  of this approximation is  $c$ . The expected linear potential  $ELP(\Gamma_{\text{in}}, \Gamma_{\text{out}})$  of the approximation is the quadratic sum of correlations for all characteristics belonging to it.

In the linear attack implemented with this approximation, we evaluate its empirical correlation by performing partial encryption and decryption with the guessed values for some subkeys. The key candidate is accepted if its empirical correlation is greater than the predetermined value of the threshold  $\tau_L$ . Since the linear attack belongs to the statistical cryptanalysis, the two errors under the differential attack setting also exist in the linear attack setting. Let  $a$  be the advantage [24] of the attack. Then, the proportion of keys that survives after the subkey enumeration phase equals  $2^{-a}$ . Equivalently, we have  $\beta = 2^{-a}$ .

Suppose that  $N_L$  known plaintexts participate in the key-recovery attack. With the method in [9], if we set the threshold as

$$\tau_L = \sqrt{1/N_L} \cdot \Phi^{-1} \left( 1 - 2^{-(a+1)} \right),$$

and  $N_L$  is sufficiently large, the success probability of the attack is approximated by

$$P_S \approx \Phi \left( \frac{c \cdot \sqrt{N_L} - \Phi^{-1}(1 - 2^{-(a+1)}) \cdot \sqrt{1 + N_L \cdot 2^{-n}}}{\sqrt{1 + N_L \cdot (ELP(\Gamma_{in}, \Gamma_{out}) - c^2)}} \right), \quad (2)$$

where  $\Phi(\cdot)$  signifies the cumulative distribution function of the standard normal distribution.

### 3 19-Round Linear Attack on GIFT-64

This section first states the selection phase of the linear distinguisher. After that, the first linear attack on GIFT-64 is proposed.

#### 3.1 Selecting Linear Distinguishers

With the experimental result in [30], the maximum absolute value of the correlation for 13-round linear characteristics is  $2^{-34}$ . Given the weak linear hull effect of GIFT-64, we conjecture that there is no 13-round linear approximation with the expected linear potential more significant than  $2^{-64}$ . Thus, we manage to apply 12-round linear approximations to launch key-recovery attacks. Considering that GIFT-64 achieves the full diffusion after three rounds [4], we expect to append three and four rounds before and after the distinguisher, respectively.

Note that the maximum correlation of 12-round linear characteristics is  $2^{-31}$ . We first apply the SAT solver to exhaustively search for all 12-round linear trails with correlations being  $2^{-31}$  and obtain 5120 trails in total. Then, the possibilities of implementing 19(=3+12+4)-round linear attacks with these trails are evaluated. Denote  $\text{GSB}_L$  the number of subkey bits involved in the subkey enumeration phase. The distribution for the number of linear trails with distinct values of  $\text{GSB}_L$  is provided in Fig. 1. A rough estimation indicates that all 5120 trails have the potentials to launch valid key-recovery attacks. Hence, we turn attention to the  $ELP$ 's of the 5120 linear approximations containing the 5120 trails because the value of  $ELP$  affects the data requirement, which acts as a crucial criterion for the performance of the linear attack. Due to the considerable number of linear approximations, we can only exhaustively search for all trails with correlations larger than  $2^{-40}$  and give approximate estimations for the  $ELP$ 's. The distribution for the number of linear approximations with different features is presented in Table 4 of Appendix A.

The experimental results illustrated in Table 4 narrow the range of candidate distinguishers to 32 linear approximations, which are provided in Table 5 of Appendix B. If we aim at linear attacks with lower data requirements, the group of 16 linear approximations L00 - L15 with  $ELP = 2^{-61.607}$  and  $\text{GSB}_L = 101$  is preferred. The group of 16 linear approximations L16 - L31 with  $ELP = 2^{-61.611}$  and  $\text{GSB}_L = 91$  is suitable for linear attacks with lower time complexities. In the following 19-round attack, we employ L16 as the distinguisher.

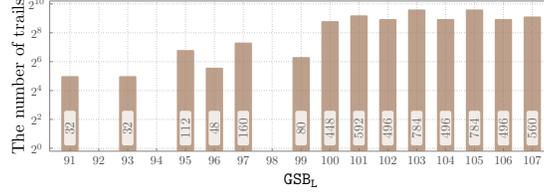


Fig. 1. Distribution for the number of 12-round linear trails with correlations  $2^{-31}$ .

### 3.2 19-Round Linear Attack on GIFT-64

The 12-round linear approximation L16 is exploited to launch a 19-round attack. To improve the accuracy of the *ELP* of the distinguisher, we apply the SAT solver to exhaustively search for all trails with correlations being larger than  $2^{-46}$ , which takes about 10 hours on one processor of a server with AMD EPYC 7302 16-Core Processor. Finally, the *ELP* of L16 is  $2^{-61.607}$ . Please refer to Appendix C for more details about L16.

In the attack, we append three and four rounds before and after the linear distinguisher, respectively. The key-recovery attack is illustrated in Fig. 2, where  $X^i$  and  $Y^i$  denote the 64-bit input and output of the SubCells operation in the  $i$ -th round ( $0 \leq i \leq 19$ ),  $EY^i$  represents the 64-bit state  $P^{-1}(X^{i+1})$ ,  $RK^i$  stands for the  $i$ -th round key, and  $EK^i$  is referred to as the equivalent round key  $P^{-1}(RK^i)$ . In the following, we use  $X^i[j]$  to represent the  $j$ -th bit of  $X^i$ .

Suppose that the number of required plaintext-ciphertext pairs is  $N_L$ . The attack is realised with the following steps.

S<sub>L</sub>1 Allocate a counter  $C_1^L[z_1]$  for each of  $2^{60}$  possible values of

$$z_1 = X^{17}[32, 34-36, 38-40, 43, 44, 46, 47] \parallel EY^{17}[0-31, 48-63] \parallel t_1,$$

where  $t_1 = X^3[20] \oplus X^3[21] \oplus X^3[28] \oplus X^3[29] \oplus X^{17}[42]$ . Then, for each possible 60-bit subkey value

$$RK^0[16-31] \parallel RK^1[12-15, 28-31] \parallel EK^{17}[17, 18, 21, 22] \parallel EK^{18}[0-31],$$

we compute the value of  $z_1$  and update  $C_1^L[z_1]$  with  $C_1^L[z_1] + 1$ . In this step, the time mainly spends on the *GS* operation, the XOR operation, and the memory access. Following the method in [26], we view one memory access to a large table as one 19-round of encryption. Thus, the dominant time complexity is  $N_L \cdot 2^{60}$  memory accesses to a table with  $2^{60}$  elements.

S<sub>L</sub>2 Allocate a counter  $C_2^L[z_2]$  for each of  $2^{56}$  possible values of

$$z_2 = X^{17}[32, 34-36, 38-40, 43, 44, 46-49, 51-53, 55-57, 59-61, 63] \parallel EY^{17}[0-31] \parallel t_1.$$

For each possible 4-bit subkey value  $EK^{17}[25, 26, 29, 30]$ , we compute the value of  $z_2$  and update  $C_2^L[z_2]$  as  $C_2^L[z_2] + C_1^L[z_1]$ . Similarly to the case in S<sub>L</sub>1, the dominant time complexity of this step is  $2^{60} \cdot 2^{60} \cdot 2^4 = 2^{124}$  memory accesses to a table with  $2^{56}$  elements.

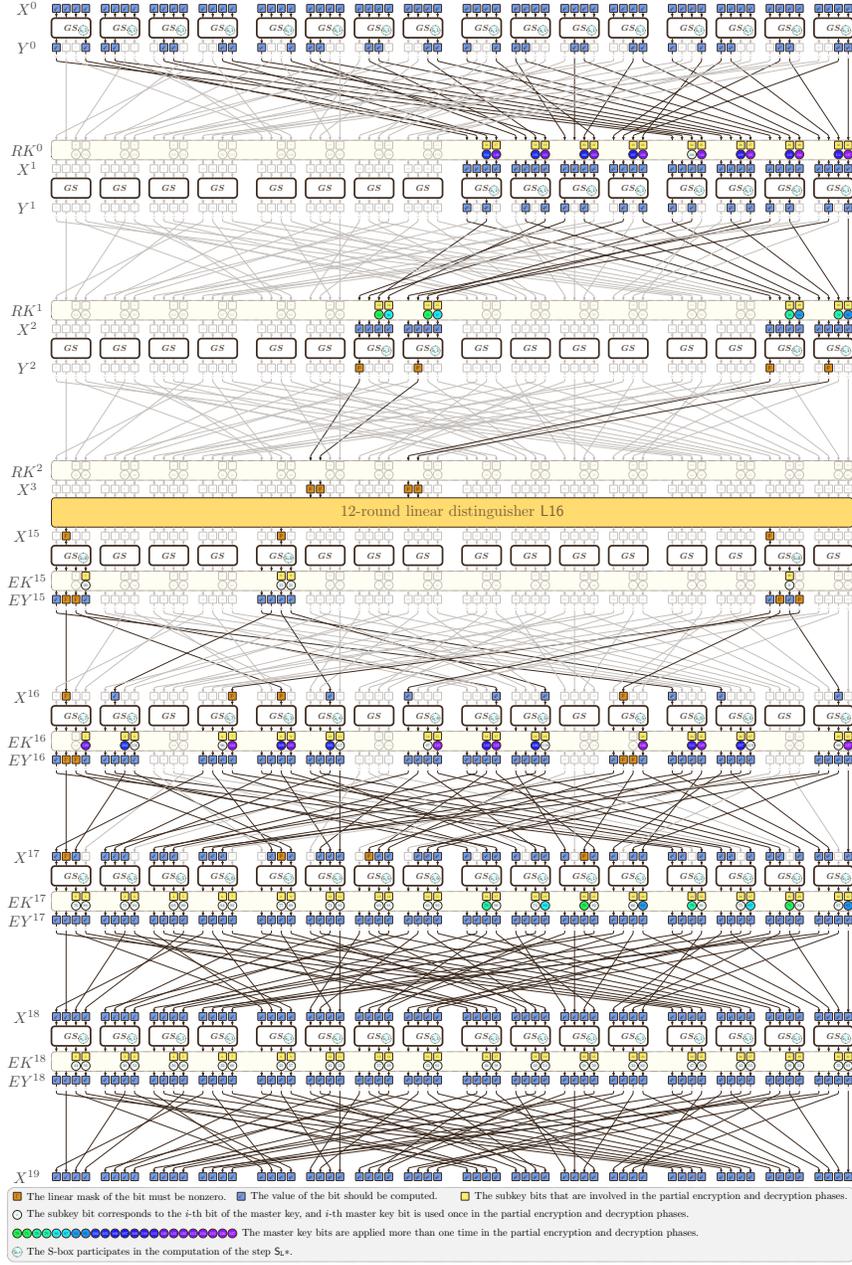


Fig. 2. Linear key-recovery attack on 19-round GIFT-64.

S<sub>L</sub>3 Allocate a counter  $C_3^L[z_3]$  for each of  $2^{50}$  possible values of

$$z_3 = X^{17}[\text{Index}^{\text{Sl}3}(X^{17})] \parallel EY^{17}[0-3, 8-19, 24-31] \parallel t_2,$$

where  $\text{Index}^{\text{Sl}3}(X^{17})$  is a index set containing the bit positions that should be memorised,

$$\text{Index}^{\text{Sl}3}(X^{17}) = \{4, 6, 21, 23, 32, 34-36, 38, 40, 43, 44, 46-49, 51, 53, 55-57, 59-61, 63\},$$

and  $t_2 = t_1 \oplus X^{16}[18]$ . For each possible 4-bit subkey value  $EK^{17}[2, 3, 10, 11]$ , we compute the value of  $z_3$  and update  $C_3^L[z_3]$  as  $C_3^L[z_3] + C_2^L[z_2]$ . The dominant time complexity of this step is  $2^{56} \cdot 2^{64} \cdot 2^4 = 2^{124}$  memory accesses to a table with  $2^{50}$  elements.

S<sub>L</sub>4 Allocate a counter  $C_4^L[z_4]$  for each of  $2^{44}$  possible values of

$$z_4 = X^{16}[22, 28] \parallel X^{17}[\text{Index}^{\text{Sl}4}(X^{17})] \parallel EY^{17}[0-3, 8-19, 24-31] \parallel t_2,$$

where

$$\text{Index}^{\text{Sl}4}(X^{17}) = \{32, 34, 35, 40, 43, 44, 46-49, 51, 56, 57, 59-61, 63\}.$$

For each possible 2-bit subkey value  $EK^{16}[11, 14]$ , we compute the value of  $z_4$  and update  $C_4^L[z_4]$  as  $C_4^L[z_4] + C_3^L[z_3]$ . The dominant time complexity of this step is  $2^{50} \cdot 2^{68} \cdot 2^2 = 2^{120}$  memory accesses to a table with  $2^{44}$  elements.

S<sub>L</sub>5 Allocate a counter  $C_5^L[z_5]$  for each of  $2^{33}$  possible values of

$$z_5 = X^{16}[22, 28, 35, 39] \parallel X^{17}[\text{Index}^{\text{Sl}5}(X^{17})] \parallel EY^{17}[0-3, 12-19, 28-31] \parallel t_3,$$

where  $\text{Index}^{\text{Sl}5}(X^{17}) = \{32, 34, 35, 44, 46-49, 51, 60, 61, 63\}$  and  $t_3 = t_2 \oplus X^{16}[45]$ . For each possible 5-bit subkey value  $EK^{16}[19] \parallel EK^{17}[4, 5, 12, 13]$ , we compute the value of  $z_5$  and update  $C_5^L[z_5]$  as  $C_5^L[z_5] + C_4^L[z_4]$ . The dominant time complexity of this step is  $2^{44} \cdot 2^{70} \cdot 2^5 = 2^{119}$  memory accesses to a table with  $2^{33}$  elements.

S<sub>L</sub>6 Allocate a counter  $C_6^L[z_6]$  for each of  $2^{22}$  possible values of

$$z_6 = X^{16}[22, 28, 35, 39, 48, 52, 62] \parallel X^{17}[32, 34, 35, 48, 49, 51] \parallel EY^{17}[0-3, 16-19] \parallel t_3.$$

For each possible 6-bit subkey value  $EK^{16}[27, 30] \parallel EK^{17}[6, 7, 14, 15]$ , we compute the value of  $z_6$  and update  $C_6^L[z_6]$  as  $C_6^L[z_6] + C_5^L[z_5]$ . The dominant time complexity of this step is  $2^{33} \cdot 2^{75} \cdot 2^6 = 2^{114}$  memory accesses to a table with  $2^{22}$  elements.

S<sub>L</sub>7 Allocate a counter  $C_7^L[z_7]$  for each of  $2^8$  possible values of

$$z_7 = X^{16}[5, 22, 28, 35, 39, 48, 52, 62] \parallel t_4,$$

where  $t_4 = t_3 \oplus X^{16}[1] \oplus X^{16}[15]$ . For each possible 6-bit subkey value  $EK^{16}[3, 6] \parallel EK^{17}[0, 1, 8, 9]$ , we compute the value of  $z_7$  and update  $C_7^L[z_7]$  as  $C_7^L[z_7] + C_6^L[z_6]$ . The number of memory accesses in this step is  $2^{22} \cdot 2^{81} \cdot 2^6 = 2^{109}$ . As the number of counters is relatively small, the time complexity of this step is not dominated by memory accesses. However, note that the number of *GS* operations and the number of XOR operations are about  $\mathcal{O}(2^{109})$ . Therefore, the time complexity of this step is bounded by  $2^{109}$  19-round of encryptions.

- S<sub>L</sub>8 Initialise a counter  $\Sigma_L$ . For each possible 4-bit subkey value  $EK^{15}[1, 8, 9, 28]$ , we compute the value of  $t_5 = t_4 \oplus X^{15}[1] \oplus X^{15}[18] \oplus X^{15}[46]$ . If  $t_5$  equals zero, we update  $\Sigma_L$  as  $\Sigma_L + C_7^L[z_7]$ . With a similar analysis as in S<sub>L</sub>7, the time complexity of this step is bounded by  $2^8 \cdot 2^{87} \cdot 2^4 = 2^{99}$  19-round of encryptions.
- S<sub>L</sub>9 We set the threshold as  $\tau_L$ . The key guess will be accepted as a candidate if the counter  $\Sigma_L$  validates the condition  $|\Sigma_L/N_L - 0.5| > \tau_L$ . Then, all master keys that are compatible with the guessed 91 subkey bits are tested exhaustively against a maximum of two plaintext-ciphertext pairs.

*Complexity Analysis.* We set the advantage of the attack as  $a = 1.40$  and the number of pairs  $N_L$  as  $2^{62.96}$ . So, the data complexity of this attack is  $2^{62.96}$ . With Eq. (2), the success probability is  $P_S = 60.00\%$ . The time complexity in each step between S<sub>L</sub>1 and S<sub>L</sub>6 depends on the number of accesses to the memory. Following the method in [26], we consider one memory access to the largest counter  $C_1^L[z_1]$  as one 19-round of encryption. The time complexity of steps S<sub>L</sub>1 - S<sub>L</sub>6 is bounded by  $(N_L \cdot 2^{60} + 2^{124} + 2^{124} + 2^{120} + 2^{119} + 2^{114})$  19-round of encryptions. The time complexity of S<sub>L</sub>9 is about  $2^{128} \cdot 2^{-a} \cdot (1 + 2^{-64})$  19-round of encryptions. Then, the time complexity of the attack is about  $2^{127.11}$  19-round of encryptions. Since  $C_1^L[z_1]$  constitutes the most remarkable memory, the memory complexity is roughly  $2^{60}$ . Given that the time complexity of the 19-round linear attack is  $2^{127.11}$ , we claim that the success probability of the attack is 60.00%, and it cannot be improved by repeating the entire work as the time complexity will go beyond  $2^{128}$ .

## 4 Differential Attack Without Using the Full Codebook

In [13], Chen et al. proposed a 20-round differential attack on GIFT-64 with the full codebook. We aim at improving this cryptanalytic result in this section.

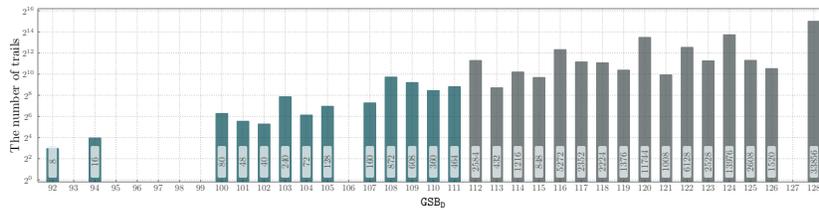
### 4.1 Selecting Differential Distinguishers

A common countermeasure to reduce the data complexity in the differential attack is to construct structures. Since GIFT-64 does not employ any whitening key at the input, we can create structures at the output of the first SubCells operation. Given the full diffusion of GIFT-64 with three rounds, we infer that the maximum number of rounds annexed before the differential distinguisher in the attack is three. On the other side, note that the 20-round attack in [13] attached four rounds after the distinguisher. Therefore, we also expect to append at least four rounds after the distinguisher.

In the selection phase of differential distinguishers, we first set the objective probability as a relatively high value and obtain candidate trails with the SAT solver. Then, we check the possibility of using the trail at hand to launch a valid attack. If none of the existing trails actualises a feasible attack, we lower the objective probability in the automatic search and repeat the abovementioned

procedures. The selection task terminates until we identify at least one proper distinguisher.

Based on previous analyses on GIFT-64 [13,30], we know that the longest differential trail that can be utilised in attacks covers 13-round, and the probability of the optimal 13-round trail achieves  $2^{-62}$ . Hence, we first fix the objective differential probability in the automatic search as  $2^{-62}$  and discover 288 trails possessing the maximum probability. However, when these trails are exploited to launch  $20(= 3 + 13 + 4)$ -round differential attacks, the time complexities go beyond  $2^{128}$  for the extensively involved subkey bits in the subkey enumeration phase. In the following, we use  $\text{GSB}_D$  to stand for the number of subkey bits involved in the subkey enumeration phase of the differential attack. Next, we reduce the objective differential probability and discover no trial with probability being  $2^{-62.415}$  or  $2^{-62.83}$ . Further, the objective probability is turned down to  $2^{-63}$ , and the SAT solver returns 5184 trails. Again, after checking all the 5184 trails, we find that none of them facilitates a valid 20-round attack for the considerable time complexity. Subsequently, we get 6272 trails with probabilities being  $2^{-63.415}$  and also notice that all the 6272 trails face the risk of enormous time complexity. Since the SAT solver does not identify any trail with probability being  $2^{-63.83}$ , we lower the objective probability to  $2^{-64}$ . 92768 trails are returned, and the distribution for the number of trails with different values of  $\text{GSB}_D$  is exhibited in Fig. 3. A rough investigation shows that the 3096 trails with  $\text{GSB}_D < 112$  are qualified for 20-round valid differential attacks.



**Fig. 3.** Distribution for the number of differential trails with probabilities  $2^{-64}$ .

After controlling the time complexity of the attack, we focus on the data complexity, which is affected by the probability of the distinguisher. Before evaluating the probability of the distinguisher, we notice that some of the 3096 candidate trails share the same input and output differences, and the number of distinct differentials is 2392. In order to obtain approximate evaluations for the probabilities of the 2392 differentials, we use the SAT solver to search for all differential trails within each differential with probabilities being larger than  $2^{-71}$ . The distribution for the number of differentials with distinct probabilities can be found in Table 2. Then, we narrow the range of candidate distinguishers to the 32 differentials with probabilities being  $2^{-61.313}$ , which are listed in



structures at the position of  $Y^0$ . In each structure, the 16 bits

$$Y^0[1, 6, 11, 12, 17, 22, 27, 28, 33, 38, 43, 44, 49, 54, 59, 60]$$

with the difference being zero in Fig. 6 are fixed, and the values of the remaining 48 bits are traversed. Then, according to the 4-bit value  $Y^0[32, 37, 40, 45]$ , the elements in the structure are further partitioned into 16 groups  $\mathcal{G}_{0x0}, \mathcal{G}_{0x1}, \dots, \mathcal{G}_{0xf}$ , and all elements  $Y^0$  in  $\mathcal{G}_i$  validate the equation  $Y^0[32, 37, 40, 45] = i$ . After that, one pair is generated by respectively drawing one element from two groups  $\mathcal{G}_i$  and  $\mathcal{G}_j$  with  $i \oplus j = 0xf$ . Thus,  $2^{91}$  pairs can be created with one structure composed of  $2^{48}$  elements.

In the attack, we prepare  $\mathcal{S}$  structures and obtain  $N_1 = \mathcal{S} \cdot 2^{91}$  pairs. In this way, the data complexity of the attack is  $\mathcal{S} \cdot 2^{48}$ . For each pair  $(Y^0, Y'^0)$ , we compute the values of the plaintexts  $(P, P')$  by applying  $GS^{-1}$  to every nibble of the two states  $(Y^0, Y'^0)$ . By querying the oracle, we obtain the corresponding values of the ciphertexts  $(C, C')$ . To minimise the time complexity in the subsequent subkey enumeration phase, we also consider the property of the key schedule.

In the first step, we guess the value of  $RK^0[10, 11]$  and check whether the 4-bit difference validates  $\Delta Y^1[16] = \Delta Y^1[17] = \Delta Y^1[18] = \Delta Y^1[19] = 0$ . The remaining  $N_1 \cdot 2^{-4}$  pairs will participate in the following processes. We repeat this guess-and-check procedure for the remaining 28-bit of  $RK^0[8, 9, 12-31] \parallel RK^1[18-23]$  involved in the partial encryption phase. The time complexity and the number of remaining pairs in steps  $S_D1 - S_D42$  illustrated in Fig. 6 are detailed in Table 3. After enumerating the related bits in  $RK^0$  and  $RK^1$ , we obtain  $N_D \triangleq N_1 \cdot 2^{-44}$  pairs that match the input difference of the 13-round distinguisher. Then, we turn to the tail of the distinguisher. The order to enumerate the subkey is selected in order to filter out the pairs that cannot result in the right pairs as soon as possible.

We set a counter to record the number of right pairs that validate the input and output differences of the 13-round distinguisher. With the analysis in Table 3, for random key guesses, the number of right pairs is about  $N_1 \cdot 2^{-108}$ . For the right key guess, the number of right pairs is expected to be  $N_1 \cdot 2^{-44} \cdot 2^{-61.31} = 2^{-105.31}$ . Thus, the number of right pairs follows a binomial distribution with parameters  $(N_D, p_0 = 2^{-61.31})$  in the case of the good key and  $(N_D, p_1 = 2^{-64})$  otherwise. The threshold is fixed as  $\tau_D$ , and the key guess will be accepted as a candidate if the counter of right pairs is no less than  $\tau_D$ . For all surviving candidates for the 107-bit subkey involved in the subkey enumeration phase, we exhaustively search for the value of the remaining 21-bit with at most two plaintext-ciphertext pairs.

*Complexity Analysis.* From Table 3, we know the time complexity  $T_1$  regarding the subkey enumeration phase is about  $N_1 \cdot 2^{23.53} \cdot \frac{1}{20 \cdot 16} \approx N_1 \cdot 2^{15.21}$  20-round of encryptions. The time complexity  $T_2$  to exhaustively check the value of the remaining 21-bit master key is  $2^{128} \cdot \beta \cdot (1 - 2^{-64})$  20-round of encryptions. We set the threshold as  $\tau_D = 1$  and the number of structures as  $\mathcal{S} = 2^{14.58}$ .

So, the data requirement of the attack is  $2^{62.58}$  chosen plaintexts. With the method recalled in Sect. 2.3, the time complexity of this attack is  $2^{125.50}$ , and the success probability is  $P_S = 70.00\%$ . Since we should memorise the right pairs, the memory complexity of this attack is roughly  $2^{62.58}$ .

## 5 Conclusion

This work is motivated by filling the vacancy of the linear attack on GIFT-64. Firstly, we apply the automatic method to search for linear approximations

**Table 3.** Detailed computation of complexity.

Step	Guessed subkey bits	Condition on the difference of the state	#(Remaining pairs)	Time complexity (GS operations)
S <sub>01</sub>	$RK^0[10, 11]$	$\Delta Y^1[16] = \Delta Y^1[17] = \Delta Y^1[18] = \Delta Y^1[19] = 0$	$N_1 \cdot 2^{-4}$	$2 \cdot N_1 \cdot 2^2$
S <sub>02</sub>	$RK^0[26, 27]$	$\Delta Y^1[52] = \Delta Y^1[53] = \Delta Y^1[54] = \Delta Y^1[55] = 0$	$N_1 \cdot 2^{-4} \cdot 2^{-4}$	$2 \cdot N_1 \cdot 2^{-4} \cdot 2^2 \cdot 2^2$
S <sub>03</sub>	$RK^0[16, 17]$	$\Delta Y^1[32] = \Delta Y^1[33] = \Delta Y^1[34] = 0$	$N_1 \cdot 2^{-8} \cdot 2^{-3}$	$2 \cdot N_1 \cdot 2^{-8} \cdot 2^4 \cdot 2^2$
S <sub>04</sub>	$RK^0[18, 19]$	$\Delta Y^1[37] = \Delta Y^1[38] = \Delta Y^1[39] = 0$	$N_1 \cdot 2^{-11} \cdot 2^{-3}$	$2 \cdot N_1 \cdot 2^{-11} \cdot 2^6 \cdot 2^2$
S <sub>05</sub>	$RK^0[20, 21]$	$\Delta Y^1[40] = \Delta Y^1[42] = \Delta Y^1[43] = 0$	$N_1 \cdot 2^{-14} \cdot 2^{-3}$	$2 \cdot N_1 \cdot 2^{-14} \cdot 2^8 \cdot 2^2$
S <sub>06</sub>	$RK^0[22, 23]$	$\Delta Y^1[44] = \Delta Y^1[45] = \Delta Y^1[47] = 0$	$N_1 \cdot 2^{-17} \cdot 2^{-3}$	$2 \cdot N_1 \cdot 2^{-17} \cdot 2^{10} \cdot 2^2$
S <sub>07</sub>	$RK^1[20, 21]$	$\Delta Y^2[40] = \Delta Y^2[41] = \Delta Y^2[42] = \Delta Y^2[43] = 0$	$N_1 \cdot 2^{-20} \cdot 2^{-4}$	$2 \cdot N_1 \cdot 2^{-20} \cdot 2^{12} \cdot 2^2$
S <sub>08</sub>	$RK^0[8, 9]$	$\Delta Y^1[16] = \Delta Y^1[17] = \Delta Y^1[18] = 0$	$N_1 \cdot 2^{-24} \cdot 2^{-3}$	$2 \cdot N_1 \cdot 2^{-24} \cdot 2^{14} \cdot 2^2$
S <sub>09</sub>	$RK^0[14, 15]$	$\Delta Y^1[28] = \Delta Y^1[29] = \Delta Y^1[31] = 0$	$N_1 \cdot 2^{-27} \cdot 2^{-3}$	$2 \cdot N_1 \cdot 2^{-27} \cdot 2^{16} \cdot 2^2$
S <sub>010</sub>	$RK^0[24, 25]$	$\Delta Y^1[48] = \Delta Y^1[49] = \Delta Y^1[50] = 0$	$N_1 \cdot 2^{-30} \cdot 2^{-3}$	$2 \cdot N_1 \cdot 2^{-30} \cdot 2^{18} \cdot 2^2$
S <sub>011</sub>	$RK^0[30, 31]$	$\Delta Y^1[60] = \Delta Y^1[61] = \Delta Y^1[63] = 0$	$N_1 \cdot 2^{-33} \cdot 2^{-3}$	$2 \cdot N_1 \cdot 2^{-33} \cdot 2^{20} \cdot 2^2$
S <sub>012</sub>	$RK^0[12, 13]$	$\Delta Y^1[24] = \Delta Y^1[25] = \Delta Y^1[26] = \Delta Y^1[27] = 0$	$N_1 \cdot 2^{-36} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-36} \cdot 2^{22} \cdot 2^2$
S <sub>013</sub>	$RK^0[28, 29]$	$\Delta Y^1[56] = \Delta Y^1[57] = \Delta Y^1[58] = \Delta Y^1[59] = 0$	$N_1 \cdot 2^{-38} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-38} \cdot 2^{24} \cdot 2^2$
S <sub>014</sub>	$RK^1[18, 19]$	$\Delta Y^2[36] = \Delta Y^2[37] = \Delta Y^2[38] = \Delta Y^2[39] = 0$	$N_1 \cdot 2^{-40} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-40} \cdot 2^{26} \cdot 2^2$
S <sub>015</sub>	$RK^1[22, 23]$	$\Delta Y^2[44] = \Delta Y^2[45] = \Delta Y^2[46] = \Delta Y^2[47] = 0$	$N_1 \cdot 2^{-42} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-42} \cdot 2^{28} \cdot 2^2$
S <sub>016</sub>	$EK^{18}[0, 1] \parallel EK^{19}[0, 1, 8, 9, 16, 17, 24, 25]$	$\Delta X^{18}[1] = \Delta X^{18}[3] = 0$	$N_1 \cdot 2^{-44} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-44} \cdot 2^{30} \cdot 2^{10} \cdot 5$
S <sub>017</sub>	$EK^{18}[2, 3]$	$\Delta X^{18}[5] = \Delta X^{18}[7] = 0$	$N_1 \cdot 2^{-46} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-46} \cdot 2^{40} \cdot 2^2$
S <sub>018</sub>	$EK^{18}[4, 5]$	$\Delta X^{18}[9] = \Delta X^{18}[11] = 0$	$N_1 \cdot 2^{-48} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-48} \cdot 2^{42} \cdot 2^2$
S <sub>019</sub>	$EK^{18}[6, 7]$	$\Delta X^{18}[13] = \Delta X^{18}[15] = 0$	$N_1 \cdot 2^{-50} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-50} \cdot 2^{44} \cdot 2^2$
S <sub>020</sub>	$EK^{18}[8, 9] \parallel EK^{19}[2, 3, 10, 11, 18, 19, 26, 27]$	$\Delta X^{18}[16] = \Delta X^{18}[18] = 0$	$N_1 \cdot 2^{-52} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-52} \cdot 2^{46} \cdot 2^{10} \cdot 5$
S <sub>021</sub>	$EK^{18}[10, 11]$	$\Delta X^{18}[20] = \Delta X^{18}[22] = 0$	$N_1 \cdot 2^{-54} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-54} \cdot 2^{56} \cdot 2^2$
S <sub>022</sub>	$EK^{18}[12, 13]$	$\Delta X^{18}[24] = \Delta X^{18}[26] = 0$	$N_1 \cdot 2^{-56} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-56} \cdot 2^{58} \cdot 2^2$
S <sub>023</sub>	$EK^{18}[14, 15]$	$\Delta X^{18}[28] = \Delta X^{18}[30] = 0$	$N_1 \cdot 2^{-58} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-58} \cdot 2^{60} \cdot 2^2$
S <sub>024</sub>	$EK^{18}[16, 17] \parallel EK^{19}[4, 5, 12, 13, 20, 21, 28, 29]$	$\Delta X^{18}[33] = \Delta X^{18}[35] = 0$	$N_1 \cdot 2^{-60} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-60} \cdot 2^{62} \cdot 2^{10} \cdot 5$
S <sub>025</sub>	$EK^{18}[18, 19]$	$\Delta X^{18}[37] = \Delta X^{18}[39] = 0$	$N_1 \cdot 2^{-62} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-62} \cdot 2^{72} \cdot 2^2$
S <sub>026</sub>	$EK^{18}[20, 21]$	$\Delta X^{18}[41] = \Delta X^{18}[43] = 0$	$N_1 \cdot 2^{-64} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-64} \cdot 2^{74} \cdot 2^2$
S <sub>027</sub>	$EK^{18}[22, 23]$	$\Delta X^{18}[45] = \Delta X^{18}[47] = 0$	$N_1 \cdot 2^{-66} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-66} \cdot 2^{76} \cdot 2^2$
S <sub>028</sub>	$EK^{18}[28, 29] \parallel EK^{19}[6, 7, 14, 15, 22, 23, 30, 31]$	$\Delta X^{18}[56] = \Delta X^{18}[58] = 0$	$N_1 \cdot 2^{-68} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-68} \cdot 2^{78} \cdot 2^{10} \cdot 5$
S <sub>029</sub>	$EK^{17}[19]$	$\Delta X^{17}[36] = \Delta X^{17}[37] = \Delta X^{17}[38] = \Delta X^{17}[39] = 0$	$N_1 \cdot 2^{-70} \cdot 2^{-4}$	$2 \cdot N_1 \cdot 2^{-70} \cdot 2^{88} \cdot 2^4$
S <sub>030</sub>	$EK^{18}[26, 27]$	$\Delta X^{18}[52] = \Delta X^{18}[54] = 0$	$N_1 \cdot 2^{-74} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-74} \cdot 2^{89} \cdot 2^2$
S <sub>031</sub>	$EK^{17}[11]$	$\Delta X^{17}[20] = \Delta X^{17}[21] = \Delta X^{17}[23] = 0$	$N_1 \cdot 2^{-76} \cdot 2^{-3}$	$2 \cdot N_1 \cdot 2^{-76} \cdot 2^{91} \cdot 2^4$
S <sub>032</sub>	$EK^{18}[30, 31]$	$\Delta X^{18}[60] = \Delta X^{18}[62] = 0$	$N_1 \cdot 2^{-79} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-79} \cdot 2^{92} \cdot 2^2$
S <sub>033</sub>	$EK^{17}[27]$	$\Delta X^{17}[53] = \Delta X^{17}[54] = \Delta X^{17}[55] = 0$	$N_1 \cdot 2^{-81} \cdot 2^{-3}$	$2 \cdot N_1 \cdot 2^{-81} \cdot 2^{94} \cdot 2^4$
S <sub>034</sub>	$EK^{18}[24, 25]$	$\Delta X^{18}[48] = \Delta X^{18}[50] = 0$	$N_1 \cdot 2^{-84} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-84} \cdot 2^{95} \cdot 2^2$
S <sub>035</sub>	$EK^{17}[2, 3]$	$\Delta X^{17}[4] = \Delta X^{17}[6] = \Delta X^{17}[7] = 0$	$N_1 \cdot 2^{-86} \cdot 2^{-3}$	$2 \cdot N_1 \cdot 2^{-86} \cdot 2^{97} \cdot 2^2$
S <sub>036</sub>	-	$\Delta X^{16}[16] = \Delta X^{16}[17] = \Delta X^{16}[18] = \Delta X^{16}[19] = 0$	$N_1 \cdot 2^{-89} \cdot 2^{-3}$	$2 \cdot N_1 \cdot 2^{-89} \cdot 2^{99}$
S <sub>037</sub>	$EK^{17}[6, 7]$	$\Delta X^{17}[12] = \Delta X^{17}[15] = 0$	$N_1 \cdot 2^{-92} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-92} \cdot 2^{99} \cdot 2^2$
S <sub>038</sub>	$EK^{17}[14, 15]$	$\Delta X^{17}[28] = \Delta X^{17}[29] = 0$	$N_1 \cdot 2^{-94} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-94} \cdot 2^{101} \cdot 2^2$
S <sub>039</sub>	$EK^{17}[22, 23]$	$\Delta X^{17}[45] = \Delta X^{17}[46] = 0$	$N_1 \cdot 2^{-96} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-96} \cdot 2^{103} \cdot 2^2$
S <sub>040</sub>	$EK^{17}[30, 31]$	$\Delta X^{17}[62] = \Delta X^{17}[63] = 0$	$N_1 \cdot 2^{-98} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-98} \cdot 2^{105} \cdot 2^2$
S <sub>041</sub>	-	$\Delta X^{16}[48] = \Delta X^{16}[49] = \Delta X^{16}[50] = \Delta X^{16}[51] = 0$	$N_1 \cdot 2^{-100} \cdot 2^{-4}$	$2 \cdot N_1 \cdot 2^{-100} \cdot 2^{107}$
S <sub>042</sub>	-	$\Delta X^{16}[52] = \Delta X^{16}[53] = \Delta X^{16}[54] = \Delta X^{16}[55] = 0$	$N_1 \cdot 2^{-104} \cdot 2^{-4}$	$2 \cdot N_1 \cdot 2^{-104} \cdot 2^{107}$
Total	-	-	-	$N_1 \cdot 2^{23.53}$

of the cipher and discover several 12-round linear distinguishers involving the minimum number of subkey bits in the subkey enumeration phase. One of these linear approximations is utilised to launch a 19-round linear attack, which is the first linear attack result on GIFT-64. In parallel, we notice that the previous differential attack of GIFT-64 covering 20 rounds claims the full codebook. To reduce the data complexity of the 20-round attack, we apply the automatic method to exhaustively check 13-round differential trails with probabilities no less than  $2^{-64}$ . A group of 32 differentials with the maximum probability is identified. One of the candidate differentials involving the minimum number of guessed subkey bits in the subkey enumeration phase is employed to realise the first 20-round differential attack without relying on the entire codebook. Given the newly proposed results, we conjecture that the resistances of GIFT-64 against differential and linear attacks do not have a significant gap.

**Acknowledgements.** The authors would like to thank the shepherd Kalikinkar Mandal and the anonymous reviewers for their valuable comments and suggestions to improve the quality of the paper. The authors also would like to thank Yong Fu for the kind discussion. The research leading to these results has received funding from the National Natural Science Foundation of China (Grant No. 62002201, Grant No. 62032014), the National Key Research and Development Program of China (Grant No. 2018YFA0704702), the Major Scientific and Technological Innovation Project of Shandong Province, China (Grant No. 2019JZZY010133), the Major Basic Research Project of Natural Science Foundation of Shandong Province, China (Grant No. ZR202010220025), and the Qingdao Postdoctor Application Research Project (Grant No. 61580070311101).

## References

1. Ankele, R., Kölbl, S.: Mind the gap - A closer look at the security of block ciphers against differential cryptanalysis. In: Selected Areas in Cryptography - SAC 2018 - 25th International Conference, Calgary, AB, Canada, August 15-17, 2018, Revised Selected Papers. pp. 163–190 (2018). [https://doi.org/10.1007/978-3-030-10970-7\\_8](https://doi.org/10.1007/978-3-030-10970-7_8)
2. Banik, S., Bogdanov, A., Peyrin, T., Sasaki, Y., Sim, S.M., Tischhauser, E., Todo, Y.: SUNDAE-GIFT. Submission to Round 1 (2019)
3. Banik, S., Chakraborti, A., Iwata, T., Minematsu, K., Nandi, M., Peyrin, T., Sasaki, Y., Sim, S.M., Todo, Y.: GIFT-COFB. IACR Cryptol. ePrint Arch. **2020**, 738 (2020), <https://eprint.iacr.org/2020/738>
4. Banik, S., Pandey, S.K., Peyrin, T., Sasaki, Y., Sim, S.M., Todo, Y.: GIFT: A small present - towards reaching the limit of lightweight encryption. In: Cryptographic Hardware and Embedded Systems - CHES 2017 - 19th International Conference, Taipei, Taiwan, September 25-28, 2017, Proceedings. pp. 321–345 (2017). [https://doi.org/10.1007/978-3-319-66787-4\\_16](https://doi.org/10.1007/978-3-319-66787-4_16)
5. Beaulieu, R., Shors, D., Smith, J., Treatman-Clark, S., Weeks, B., Wingers, L.: The SIMON and SPECK families of lightweight block ciphers. IACR Cryptol. ePrint Arch. **2013**, 404 (2013)

6. Beierle, C., Jean, J., Kölbl, S., Leander, G., Moradi, A., Peyrin, T., Sasaki, Y., Sasdrich, P., Sim, S.M.: The SKINNY family of block ciphers and its low-latency variant MANTIS. In: *Advances in Cryptology - CRYPTO 2016 - 36th Annual International Cryptology Conference*, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part II. pp. 123–153 (2016). [https://doi.org/10.1007/978-3-662-53008-5\\_5](https://doi.org/10.1007/978-3-662-53008-5_5)
7. Biham, E., Shamir, A.: Differential cryptanalysis of DES-like cryptosystems. In: *Advances in Cryptology - CRYPTO '90, 10th Annual International Cryptology Conference*, Santa Barbara, California, USA, August 11-15, 1990, Proceedings. pp. 2–21 (1990). [https://doi.org/10.1007/3-540-38424-3\\_1](https://doi.org/10.1007/3-540-38424-3_1)
8. Blondeau, C., Gérard, B., Tillich, J.: Accurate estimates of the data complexity and success probability for various cryptanalyses. *Des. Codes Cryptogr.* **59**(1-3), 3–34 (2011). <https://doi.org/10.1007/s10623-010-9452-2>
9. Blondeau, C., Nyberg, K.: Joint data and key distribution of simple, multiple, and multidimensional linear cryptanalysis test statistic and its impact to data complexity. *Des. Codes Cryptogr.* **82**(1-2), 319–349 (2017). <https://doi.org/10.1007/s10623-016-0268-6>
10. Bogdanov, A., Knudsen, L.R., Leander, G., Paar, C., Poschmann, A., Robshaw, M.J.B., Seurin, Y., Vikkelsoe, C.: PRESENT: An ultra-lightweight block cipher. In: *Cryptographic Hardware and Embedded Systems - CHES 2007, 9th International Workshop*, Vienna, Austria, September 10-13, 2007, Proceedings. pp. 450–466 (2007). [https://doi.org/10.1007/978-3-540-74735-2\\_31](https://doi.org/10.1007/978-3-540-74735-2_31)
11. Chakraborti, A., Datta, N., Jha, A., Lopez, C.M., Nandi, M., Sasaki, Y.: LOTUS-AEAD and LOCUS-AEAD. Submission to the NIST Lightweight Cryptography project (2019)
12. Chakraborti, A., Datta, N., Jha, A., Nandi, M.: HYENA. Submission to the NIST Lightweight Cryptography project (2019)
13. Chen, H., Zong, R., Dong, X.: Improved differential attacks on GIFT-64. In: *Information and Communications Security - 21st International Conference, ICICS 2019, Beijing, China, December 15-17, 2019, Revised Selected Papers*. pp. 447–462 (2019). [https://doi.org/10.1007/978-3-030-41579-2\\_26](https://doi.org/10.1007/978-3-030-41579-2_26)
14. Cho, J.Y.: Linear cryptanalysis of reduced-round PRESENT. In: Pieprzyk, J. (ed.) *Topics in Cryptology - CT-RSA 2010, The Cryptographers' Track at the RSA Conference 2010*, San Francisco, CA, USA, March 1-5, 2010. Proceedings. *Lecture Notes in Computer Science*, vol. 5985, pp. 302–317. Springer (2010). [https://doi.org/10.1007/978-3-642-11925-5\\_21](https://doi.org/10.1007/978-3-642-11925-5_21)
15. Cook, S.A.: The complexity of theorem-proving procedures. In: *Proceedings of the 3rd Annual ACM Symposium on Theory of Computing*, May 3-5, 1971, Shaker Heights, Ohio, USA. pp. 151–158 (1971). <https://doi.org/10.1145/800157.805047>
16. Ji, F., Zhang, W., Zhou, C., Ding, T.: Improved (related-key) differential cryptanalysis on GIFT. *IACR Cryptol. ePrint Arch.* **2020**, 1242 (2020)
17. Kölbl, S., Leander, G., Tiessen, T.: Observations on the SIMON block cipher family. In: *Advances in Cryptology - CRYPTO 2015 - 35th Annual Cryptology Conference*, Santa Barbara, CA, USA, August 16-20, 2015, Proceedings, Part I. pp. 161–185 (2015). [https://doi.org/10.1007/978-3-662-47989-6\\_8](https://doi.org/10.1007/978-3-662-47989-6_8)
18. Lai, X., Massey, J.L., Murphy, S.: Markov ciphers and differential cryptanalysis. In: Davies, D.W. (ed.) *Advances in Cryptology - EUROCRYPT '91, Workshop on the Theory and Application of Cryptographic Techniques*, Brighton, UK, April 8-11, 1991, Proceedings. *Lecture Notes in Computer Science*, vol. 547, pp. 17–38. Springer (1991). [https://doi.org/10.1007/3-540-46416-6\\_2](https://doi.org/10.1007/3-540-46416-6_2)

19. Liu, Y., Sasaki, Y.: Related-key boomerang attacks on GIFT with automated trail search including BCT effect. In: Information Security and Privacy - 24th Australasian Conference, ACISP 2019, Christchurch, New Zealand, July 3-5, 2019, Proceedings. pp. 555–572 (2019). [https://doi.org/10.1007/978-3-030-21548-4\\_30](https://doi.org/10.1007/978-3-030-21548-4_30)
20. Liu, Y., Wang, Q., Rijmen, V.: Automatic search of linear trails in ARX with applications to SPECK and Chaskey. In: Applied Cryptography and Network Security - 14th International Conference, ACNS 2016, Guildford, UK, June 19-22, 2016. Proceedings. pp. 485–499 (2016). [https://doi.org/10.1007/978-3-319-39555-5\\_26](https://doi.org/10.1007/978-3-319-39555-5_26)
21. Matsui, M.: Linear cryptanalysis method for DES cipher. In: Advances in Cryptology - EUROCRYPT '93, Workshop on the Theory and Application of Cryptographic Techniques, Lofthus, Norway, May 23-27, 1993, Proceedings. pp. 386–397 (1993). [https://doi.org/10.1007/3-540-48285-7\\_33](https://doi.org/10.1007/3-540-48285-7_33)
22. Rickmann, S.: Logic friday (version 1.1. 3)[computer software] (2011)
23. Sasaki, Y.: Integer linear programming for three-subset meet-in-the-middle attacks: Application to GIFT. In: Inomata, A., Yasuda, K. (eds.) Advances in Information and Computer Security - 13th International Workshop on Security, IWSEC 2018, Sendai, Japan, September 3-5, 2018, Proceedings. Lecture Notes in Computer Science, vol. 11049, pp. 227–243. Springer (2018). [https://doi.org/10.1007/978-3-319-97916-8\\_15](https://doi.org/10.1007/978-3-319-97916-8_15)
24. Selçuk, A.A.: On probability of success in linear and differential cryptanalysis. J. Cryptol. **21**(1), 131–147 (2008). <https://doi.org/10.1007/s00145-007-9013-7>
25. Sinz, C.: Towards an optimal CNF encoding of Boolean cardinality constraints. In: Principles and Practice of Constraint Programming - CP 2005, 11th International Conference, CP 2005, Sitges, Spain, October 1-5, 2005, Proceedings. pp. 827–831 (2005). [https://doi.org/10.1007/11564751\\_73](https://doi.org/10.1007/11564751_73)
26. Soleimany, H., Nyberg, K.: Zero-correlation linear cryptanalysis of reduced-round LBlock. Des. Codes Cryptogr. **73**(2), 683–698 (2014). <https://doi.org/10.1007/s10623-014-9976-y>
27. Song, L., Huang, Z., Yang, Q.: Automatic differential analysis of ARX block ciphers with application to SPECK and LEA. In: Information Security and Privacy - 21st Australasian Conference, ACISP 2016, Melbourne, VIC, Australia, July 4-6, 2016, Proceedings, Part II. pp. 379–394 (2016). [https://doi.org/10.1007/978-3-319-40367-0\\_24](https://doi.org/10.1007/978-3-319-40367-0_24)
28. Soos, M., Nohl, K., Castelluccia, C.: Extending SAT solvers to cryptographic problems. In: Kullmann, O. (ed.) Theory and Applications of Satisfiability Testing - SAT 2009, 12th International Conference, SAT 2009, Swansea, UK, June 30 - July 3, 2009. Proceedings. Lecture Notes in Computer Science, vol. 5584, pp. 244–257. Springer (2009). [https://doi.org/10.1007/978-3-642-02777-2\\_24](https://doi.org/10.1007/978-3-642-02777-2_24)
29. Sun, L., Wang, W., Wang, M.: More accurate differential properties of LED64 and Midori64. IACR Trans. Symmetric Cryptol. **2018**(3), 93–123 (2018). <https://doi.org/10.13154/tosc.v2018.i3.93-123>
30. Sun, L., Wang, W., Wang, M.: Accelerating the search of differential and linear characteristics with the SAT method. IACR Trans. Symmetric Cryptol. **2021**(1), 269–315 (2021). <https://doi.org/10.46586/tosc.v2021.i1.269-315>
31. Zhu, B., Dong, X., Yu, H.: MILP-based differential attack on round-reduced GIFT. In: Matsui, M. (ed.) Topics in Cryptology - CT-RSA 2019 - The Cryptographers' Track at the RSA Conference 2019, San Francisco, CA, USA, March 4-8, 2019, Proceedings. Lecture Notes in Computer Science, vol. 11405, pp. 372–390. Springer (2019). [https://doi.org/10.1007/978-3-030-12612-4\\_19](https://doi.org/10.1007/978-3-030-12612-4_19)

## A Distribution for the 5120 Linear Approximation

The distribution for the number of linear approximations with different features is presented in Table 4.

**Table 4.** Distribution for the number of linear approximations with different features.

		GSB <sub>L</sub> of the linear approximation													
		91	93	95	96	97	99	100	101	102	103	104	105	106	107
<i>ELP</i> of the linear approximation	$2^{-61.607}$	0	0	0	0	0	0	0	16	16	0	0	16	16	0
	$2^{-61.609}$	0	0	0	0	0	16	16	0	0	16	16	0	0	0
	$2^{-61.610}$	0	0	0	0	0	0	0	16	0	16	0	16	0	16
	$2^{-61.611}$	16	16	0	0	0	0	0	0	0	16	0	16	0	0
	$2^{-61.761}$	0	0	0	0	0	0	16	16	16	16	16	16	16	16
	$2^{-61.764}$	0	0	0	0	0	0	16	16	16	16	16	16	16	16
	$2^{-61.765}$	0	0	32	0	32	0	0	32	0	64	0	64	0	32
	$2^{-61.847}$	0	0	16	0	16	0	0	16	0	32	0	32	0	16
	$2^{-61.848}$	0	0	16	0	16	0	0	16	0	32	0	32	0	16
	$2^{-61.904}$	0	0	0	0	0	0	0	0	0	16	0	0	0	16
	$2^{-61.906}$	0	0	0	0	0	0	0	16	0	0	0	16	0	0
	$2^{-61.910}$	16	0	0	0	0	0	0	0	0	16	0	0	0	0
	$2^{-61.913}$	0	0	0	0	0	0	0	0	16	0	0	0	16	0
	$2^{-61.914}$	0	16	0	0	0	0	0	16	0	0	0	32	0	0
	$2^{-61.917}$	0	0	0	0	0	0	16	0	0	0	16	0	0	0
	$2^{-61.918}$	0	0	0	0	0	16	0	0	0	16	0	0	0	0
	$2^{-61.919}$	0	0	0	0	0	0	0	0	0	32	0	0	0	32
	$2^{-61.920}$	0	0	0	0	0	0	0	0	32	16	0	0	32	16
	$2^{-61.921}$	0	0	0	16	16	0	16	32	32	32	32	48	32	32
	$2^{-61.922}$	0	0	16	16	16	0	16	0	0	16	32	16	0	0
	$2^{-61.923}$	0	0	32	0	48	0	32	112	96	128	32	160	96	96
	$2^{-61.924}$	0	0	0	16	16	32	128	96	80	112	144	112	80	80
	$2^{-61.925}$	0	0	0	0	0	0	64	64	96	96	64	64	96	96
	$2^{-61.926}$	0	0	0	0	0	16	128	128	96	96	128	128	96	80

## B 32 Candidate Linear Approximations

Please find the 32 candidate linear approximations in Table 5.

## C Details about L16

The number of trails belonging to L16 with different correlations is demonstrated in Fig. 5, and the dominating trail with correlation being  $2^{-31}$  is provided in Table 6.

## D 32 Candidate Differentials

Please find the 32 candidate differentials in Table 7.

## E Three Dominating Trails of D03

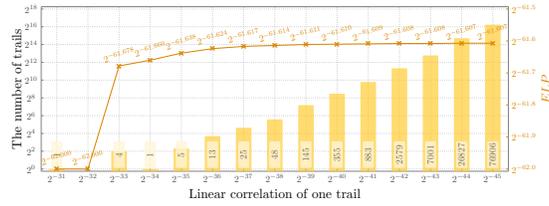
Three dominating trails of D03 with probabilities being  $2^{-64}$  can be found in Table 8.

## F Illustration for the 20-Round Differential Attack

The 20-round differential attack is demonstrated in Fig. 6.

**Table 5.** 32 candidate linear approximations.

	ID	Input mask	Output mask
Distinguishers with $ELP = 2^{-61.607}$ and $GSBL = 101$	L00	0x0000 0x0c0c 0x0000 0x0000	0x1000 0x0080 0x4000 0x2000
	L01	0x0000 0x0000 0x0000 0xc0c0	0x0400 0x0200 0x0100 0x0008
	L02	0x0000 0x0000 0x0000 0xc0c0	0x0004 0x0002 0x0001 0x0800
	L03	0x0000 0xc0c0 0x0000 0x0000	0x0100 0x0008 0x0400 0x0200
	L04	0x0000 0x0000 0x0000 0x0c0c	0x0040 0x0020 0x0010 0x8000
	L05	0x0000 0x0c0c 0x0000 0x0000	0x0010 0x8000 0x0040 0x0020
	L06	0x0000 0x0000 0x0000 0x0c0c	0x4000 0x2000 0x1000 0x0080
	L07	0x0c0c 0x0000 0x0000 0x0000	0x8000 0x0040 0x0020 0x0010
	L08	0x0000 0x0000 0x0c0c 0x0000	0x0020 0x0010 0x8000 0x0040
	L09	0x0000 0x0000 0x0c0c 0x0000	0x2000 0x1000 0x0080 0x4000
	L10	0x0c0c 0x0000 0x0000 0x0000	0x0080 0x4000 0x2000 0x1000
	L11	0x0000 0x0000 0xc0c0 0x0000	0x0200 0x0100 0x0008 0x0400
	L12	0x0000 0x0000 0xc0c0 0x0000	0x0002 0x0001 0x0800 0x0004
	L13	0xc0c0 0x0000 0x0000 0x0000	0x0800 0x0004 0x0002 0x0001
	L14	0xc0c0 0x0000 0x0000 0x0000	0x0008 0x0400 0x0200 0x0100
L15	0x0000 0xc0c0 0x0000 0x0000	0x0001 0x0800 0x0004 0x0002	
Distinguishers with $ELP = 2^{-61.611}$ and $GSBL = 91$	L16	0x0000 0x0c0c 0x0000 0x0000	0x4000 0x2000 0x0000 0x0080
	L17	0x0000 0x0000 0x0000 0x0c0c	0x0000 0x0080 0x4000 0x2000
	L18	0x0000 0xc0c0 0x0000 0x0000	0x0400 0x0200 0x0000 0x0008
	L19	0x0000 0x0000 0x0000 0xc0c0	0x0000 0x0800 0x0004 0x0002
	L20	0xc0c0 0x0000 0x0000 0x0000	0x0002 0x0000 0x0800 0x0004
	L21	0x0000 0x0000 0x0000 0xc0c0	0x0000 0x0008 0x0400 0x0200
	L22	0x0000 0xc0c0 0x0000 0x0000	0x0004 0x0002 0x0000 0x0800
	L23	0x0000 0x0000 0x0000 0x0c0c	0x0000 0x8000 0x0040 0x0020
	L24	0x0000 0x0c0c 0x0000 0x0000	0x0040 0x0020 0x0000 0x8000
	L25	0x0c0c 0x0000 0x0000 0x0000	0x0020 0x0000 0x8000 0x0040
	L26	0x0000 0x0000 0x0c0c 0x0000	0x0080 0x4000 0x2000 0x0000
	L27	0x0000 0x0000 0x0c0c 0x0000	0x8000 0x0040 0x0020 0x0000
	L28	0x0c0c 0x0000 0x0000 0x0000	0x2000 0x0000 0x0080 0x4000
	L29	0x0000 0x0000 0xc0c0 0x0000	0x0008 0x0400 0x0200 0x0000
	L30	0x0000 0x0000 0xc0c0 0x0000	0x0800 0x0004 0x0002 0x0000
	L31	0xc0c0 0x0000 0x0000 0x0000	0x0200 0x0000 0x0008 0x0400



**Fig. 5.** Distribution of trails belonging to the 12-round linear approximation L16.

**Table 6.** Dominating trail of L16 with correlation being  $2^{-31}$ .

State	Linear mask
$\Gamma X^3$	0x0000 0x0c0c 0x0000 0x0000
$\Gamma X^4$	0x0a00 0x0000 0x0a00 0x0000
$\Gamma X^5$	0x2020 0x0000 0x0000 0x0000
$\Gamma X^6$	0x5000 0x0000 0x5000 0x0000
$\Gamma X^7$	0x0000 0x2020 0x0000 0x8080
$\Gamma X^8$	0x0505 0x0000 0x0505 0x0000
$\Gamma X^9$	0xa0a0 0x0000 0xa0a0 0x0000
$\Gamma X^{10}$	0x0000 0xa0a0 0x0000 0x0000
$\Gamma X^{11}$	0x0000 0x0000 0x0000 0x0a00
$\Gamma X^{12}$	0x0002 0x0000 0x0000 0x0000
$\Gamma X^{13}$	0x8000 0x0000 0x0000 0x0000
$\Gamma X^{14}$	0x4000 0x0000 0x1000 0x0000
$\Gamma X^{15}$	0x4000 0x2000 0x0000 0x0080

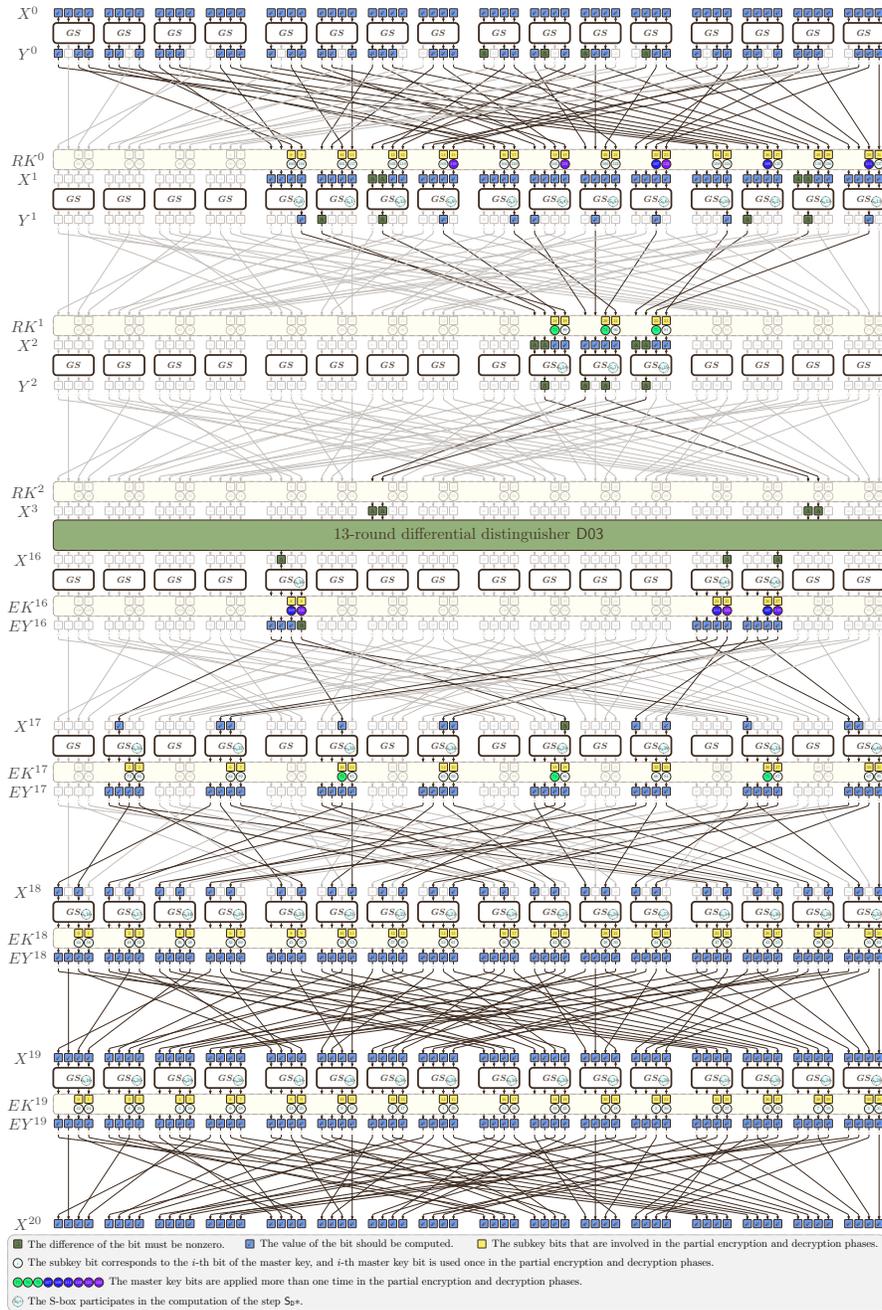


Fig. 6. Differential key-recovery attack on 20-round GIFT-64.

**Table 7.** 32 candidate differentials.

	ID	Input difference				Output difference			
Differentials with $GSB_b = 107$	D00	0x0000	0x0006	0x0000	0x000c	0x0040	0x0000	0x0011	0x0000
	D01	0x0000	0x000c	0x0000	0x0006	0x4000	0x0000	0x1100	0x0000
	D02	0x0000	0x0060	0x0000	0x00c0	0x0000	0x0040	0x0000	0x0011
	D03	0x0000	0x00c0	0x0000	0x0060	0x0000	0x4000	0x0000	0x1100
	D04	0x0000	0x0600	0x0000	0x0c00	0x0011	0x0000	0x0040	0x0000
	D05	0x0000	0x0c00	0x0000	0x0600	0x1100	0x0000	0x4000	0x0000
	D06	0x0000	0x6000	0x0000	0xc000	0x0000	0x0011	0x0000	0x0040
	D07	0x0000	0xc000	0x0000	0x6000	0x0000	0x1100	0x0000	0x4000
	D08	0x0006	0x0000	0x000c	0x0000	0x0400	0x0000	0x0110	0x0000
	D09	0x000c	0x0000	0x0006	0x0000	0x0004	0x0000	0x1001	0x0000
	D10	0x0060	0x0000	0x00c0	0x0000	0x0000	0x0400	0x0000	0x0110
	D11	0x00c0	0x0000	0x0060	0x0000	0x0000	0x0004	0x0000	0x1001
	D12	0x0600	0x0000	0x0c00	0x0000	0x0110	0x0000	0x0400	0x0000
	D13	0x0c00	0x0000	0x0600	0x0000	0x1001	0x0000	0x0004	0x0000
	D14	0x6000	0x0000	0xc000	0x0000	0x0000	0x0110	0x0000	0x0400
D15	0xc000	0x0000	0x6000	0x0000	0x0000	0x1001	0x0000	0x0004	
Differentials with $GSB_b = 110$	D16	0x0000	0x0006	0x0000	0x000c	0x4000	0x0000	0x1100	0x0000
	D17	0x0000	0x000c	0x0000	0x0006	0x0040	0x0000	0x0011	0x0000
	D18	0x0000	0x0060	0x0000	0x00c0	0x0000	0x4000	0x0000	0x1100
	D19	0x0000	0x00c0	0x0000	0x0060	0x0000	0x0040	0x0000	0x0011
	D20	0x0000	0x0600	0x0000	0x0c00	0x1100	0x0000	0x4000	0x0000
	D21	0x0000	0x0c00	0x0000	0x0600	0x0011	0x0000	0x0040	0x0000
	D22	0x0000	0x6000	0x0000	0xc000	0x0000	0x1100	0x0000	0x4000
	D23	0x0000	0xc000	0x0000	0x6000	0x0000	0x0011	0x0000	0x0040
	D24	0x0006	0x0000	0x000c	0x0000	0x0004	0x0000	0x1001	0x0000
	D25	0x000c	0x0000	0x0006	0x0000	0x0400	0x0000	0x0110	0x0000
	D26	0x0060	0x0000	0x00c0	0x0000	0x0000	0x0004	0x0000	0x1001
	D27	0x00c0	0x0000	0x0060	0x0000	0x0000	0x0400	0x0000	0x0110
	D28	0x0600	0x0000	0x0c00	0x0000	0x1001	0x0000	0x0004	0x0000
	D29	0x0c00	0x0000	0x0600	0x0000	0x0110	0x0000	0x0400	0x0000
	D30	0x6000	0x0000	0xc000	0x0000	0x0000	0x1001	0x0000	0x0004
	D31	0xc000	0x0000	0x6000	0x0000	0x0000	0x0110	0x0000	0x0400

**Table 8.** Three dominating trails with probabilities being  $2^{-64}$ .

State	The first dominating trail				The second dominating trail				The third dominating trail			
$\Delta X^3$	0x0000	0x00c0	0x0000	0x0060	0x0000	0x00c0	0x0000	0x0060	0x0000	0x00c0	0x0000	0x0060
$\Delta X^4$	0x0000	0x0000	0x0000	0x0202	0x0000	0x0000	0x0000	0x0202	0x0000	0x0000	0x0000	0x0202
$\Delta X^5$	0x000a	0x0000	0x000a	0x0000	0x0000	0x0005	0x0000	0x0005	0x0000	0x0005	0x0000	0x0005
$\Delta X^6$	0x0000	0x0000	0x0000	0x1010	0x0000	0x0000	0x0202	0x0000	0x0000	0x0000	0x0202	0x0000
$\Delta X^7$	0x0000	0x000a	0x0000	0x000a	0x0000	0x0050	0x0000	0x0050	0x0000	0x0050	0x0000	0x0050
$\Delta X^8$	0x0000	0x0000	0x0000	0x0101	0x0000	0x0000	0x0000	0x0202	0x0000	0x0000	0x0000	0x0202
$\Delta X^9$	0x000a	0x0000	0x000a	0x0000	0x0000	0x0005	0x0000	0x0005	0x000a	0x0000	0x000a	0x0000
$\Delta X^{10}$	0x0000	0x0000	0x0000	0x1010	0x0000	0x0000	0x0202	0x0000	0x0000	0x0000	0x0000	0x1010
$\Delta X^{11}$	0x0000	0x000a	0x0000	0x000a	0x0000	0x0050	0x0000	0x0050	0x0000	0x000a	0x0000	0x000a
$\Delta X^{12}$	0x0000	0x0000	0x0000	0x0101	0x0000	0x0000	0x0000	0x0202	0x0000	0x0000	0x0000	0x0101
$\Delta X^{13}$	0x000a	0x0000	0x000a	0x0000	0x000a	0x0000	0x000a	0x0000	0x000a	0x0000	0x000a	0x0000
$\Delta X^{14}$	0x0000	0x0000	0x0000	0x1010	0x0000	0x0000	0x0000	0x1010	0x0000	0x0000	0x0000	0x1010
$\Delta X^{15}$	0x0004	0x000a	0x0000	0x0000	0x0004	0x000a	0x0000	0x0000	0x0004	0x000a	0x0000	0x0000
$\Delta X^{16}$	0x0000	0x4000	0x0000	0x1100	0x0000	0x4000	0x0000	0x1100	0x0000	0x4000	0x0000	0x1100