IACR Communications in Cryptology
ISSN 3006-5496, Vol. 1, No. 1, 24 pages.

https://doi.org/10.62056/a6n5txol7

Differential-Linear Cryptanalysis of GIFT family and GIFT-based Ciphers

Shichang Wang[1,2], Meicheng Liu[1,2], Shiqi Hou[1,2] and Dongdai Lin[1,2]

[1] Key Laboratory of Cyberspace Security Defense, Institute of Information Engineering, CAS, Beijing, China

[2] School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China

Abstract. At CHES 2017, Banik et al. proposed a lightweight block cipher `GIFT` consisting of two versions `GIFT-64` and `GIFT-128`. Recently, there are lots of authenticated encryption schemes that adopt `GIFT-128` as their underlying primitive, such as `GIFT-COFB` and `HyENA`. To promote a comprehensive perception of the soundness of the designs, we evaluate their security against differential-linear cryptanalysis. For this, automatic tools have been developed to search differential-linear approximation for the ciphers based on S-boxes. With the assistance of the automatic tools, we find 13-round differential-linear approximations for `GIFT-COFB` and `HyENA`. Based on the distinguishers, 18-round key-recovery attacks are given for the message processing phase and initialization phase of both ciphers. Moreover, the resistance of `GIFT-64/128` against differential-linear cryptanalysis is also evaluated. The 12-round and 17-round differential-linear approximations are found for `GIFT-64` and `GIFT-128` respectively, which lead to 18-round and 19-round key-recovery attacks respectively. Here, we stress that our attacks do not threaten the security of these ciphers.

Keywords: Differential-linear attack · GIFT · GIFT-COFB · HyENA

1 Introduction

The past few decades have witnessed the increasingly common deployment of small computing devices, such as sensor nodes, RFID tags, smart cards, and industrial controllers, which brings a wide range of new security and privacy concerns. Since conventional cryptographic standards are not acceptable when implemented in the above highly constrained computing environment, numerous algorithms tailored for resource-constrained devices have emerged, often summarized as so-called lightweight cryptography. The lightweight block cipher family `GIFT` is designed by Banik et al. [BPP+17], which includes two versions, GIFT-64 and GIFT-128, and both have a 128-bit key size. `GIFT` inherits the design framework from `PRESENT`, with the correction of the weakness of the strong linear hull effect. In 2018, the National Institute of Standards and Technology (NIST) initiated a lightweight cryptography project to solicit, evaluate, and standardize lightweight cryptographic algorithms aiming for execution under extreme performance constraints. `GIFT-COFB` [BCI+21] instantiates the COFB (COmbined FeedBack) block cipher based Authenticated Encryption with Associated Data (AEAD) mode, using `GIFT-128` [BPP+17]. It can be implemented efficiently, and achieves desirable features, thus making its way to the finalists of NIST lightweight cryptography project. `HyENA` [CDJN19], also instantiating with `GIFT-128`, provides nonce-based authenticated encryption with associated data functionality. Here,

Supported by the National Natural Science Foundation of China (Grant No. 62122085, 12231015, 62372449, and 61936008) and the Youth Innovation Promotion Association of Chinese Academy of Sciences.

E-mail: wangshichang@iie.ac.cn (Shichang Wang), liumeicheng@iie.ac.cn (Meicheng Liu), houshiqi@iie.ac.cn (Shiqi Hou), ddlin@iie.ac.cn (Dongdai Lin)

Received: 2024-01-09 Accepted: 2024-03-05

when mentioning HyENA, we refer to its concrete instantiation based on GIFT-128, not the mode of operation. Given its salient features, like inverse-free, low XOR count, low state size, and an optimal number of nonlinear primitive calls, HyENA has been selected as one of the 32 second-round candidates of NIST lightweight cryptography project.

Unlike public-key cryptography, our confidence in the security of symmetric-key primitives mainly lies in their resistance against all known cryptanalytic methods. Therein, differential and linear cryptanalysis, introduced by Biham, Shamir [BS90] and Matsui [Mat93] respectively, are the two most profound techniques for the security evaluation of block ciphers. While the design of symmetric-key primitives assures resistance against differential and linear attacks, combining the short differential characteristics and linear approximations may be also vulnerabilities that can be exploited when evaluating their security. In 1994, Langford and Hellman [LH94] firstly showed that a differential of E_0 and a linear approximation of E_1 could be combined into a distinguisher for the entire cipher $E_1 \circ E_0$ by a technique called *differential-linear cryptanalysis* (abbreviated as DL cryptanalysis). Recently, there have been many valuable and thought-provoking developments on DL cryptanalysis. In 2017, Blondeau et al. [BLN17] developed a concise theory of differential-linear cryptanalysis by exploiting the link between differential and linear attacks. Under the assumption that the two subciphers were independent, an exact expression is given for the bias of differential-linear approximation (abbreviated as DL approximation). Bar-On et al. [BDKW19], at EUROCRYPT 2019, defined the *Differential-Linear Connectivity Table* (DLCT) to take the dependency between the two subciphers into account, and improved the differential-linear attacks on ICEPOLE and 8-round DES with DLCT. At CRYPTO 2020, Beierle et al. [BLT20] presented several improvements in the context of the differential-linear attacks on ARX ciphers and successfully applied them to Chaskey and ChaCha. Subsequently, at EUROCRYPT 2021, Coutinho and Souza Neto [CN21] proposed a new technique to find better linear approximations in ARX ciphers. At CRYPTO 2021, Liu et al. [LLL21] studied the differential-linear cryptanalysis from an algebraic perspective by introducing a technique called *Differential Algebraic Transitional Form* (DATF). Based on DATF, they developed a new theory for estimating bias and techniques for key recovery in differential-linear cryptanalysis, which were applied to Ascon, Serpent, and Grain v1. At EUROCRYPT 2021, Liu et al. [LSL21] extended the framework of DL cryptanalysis into rotational differential-linear attacks by replacing the differential part with the rotational-xor differential. As an application, they analyzed the ciphers FRIET, Xoodoo and Alzette by a practical method of evaluating the rotational differential-linear correlations for the special cases where output linear masks are unit vectors. At CRYPTO 2022, Niu et al. [NSLL22] extended the method to arbitrary output linear masks by presenting an efficient algorithm for computing (rotational) differential-linear correlation of modulo additions. Along the direction of [LSL21] and [NSLL22], Bellini et al. [BGG$^+$23] and Lv et al. [LJC23] presented automatic methods of searching differential-linear approximations for the ARX ciphers. Recently, at ASIACRYPT 2023, Hu et al. [HPTY23] revisited high-order differential-linear cryptanalysis from an algebraic perspective by extending DATF in [LLL21] into the higher-order one and successfully analysed the ciphers Ascon and Xoodyak. Despite the emergence of numerous research on differential-linear cryptanalysis, there are still many questions remaining to be solved for this analytical method, such as how to automatically search differential-linear approximation for the S-box-based ciphers. This is vital to facilitate comprehensive analysis and deepen our understanding of cryptographic designs.

1.1 Our Contributions

In this paper, we give our attention to the security of GIFT family and two GIFT-based AEADs, namely GIFT-COFB and HyENA, against differential-linear cryptanalysis. We begin with showing how to construct an automatic tool to search concisely and effectively differential-linear approximations for the S-box-based ciphers.

Automatic Tools of Searching DL Approximation for S-box-based Ciphers. An MILP (Mixed Integer Linear Programming) model has been developed to search automatically differential-linear approximations for the S-box-based ciphers. First, for an S-box, a way/algorithm is presented to derive the propagation of correlation of differential-linear approximation from its DDT (Differential Distribution Table). We have implemented the way by symbolic programming in SageMath, and correspondingly Proposition 1 is obtained which illustrates the propagation of correlation of DL approximation for the S-box of GIFT. The implementation in SageMath can be easily used to analyze other cipher's S-boxes. So the correlation of DL approximation can be efficiently computed for the S-box-based ciphers by combining with the propagation rules for other operations, such as XOR and AND. Further, an integrated model is designed to search differential-linear approximations for the common framework depicted in Figure 1. More precisely, we show how to model the differential-linear part E_m by the pattern-choosing rule, which is proved to be equivalently described by two inequalities in Theorem 1. Then the propagation of the three part E_d, E_m and E_l are merged as a whole MILP model to search differential-linear approximations. Besides, a phenomenon of the differential propagation of 3-round GIFT-128 is found, i.e. Propositions 2 and 3, which reveals the restriction of active bits in key-recovery attacks on the message processing phase of GIFT-COFB and HyENA can be directly converted into the ones of distinguisher's input. We apply our automatic tool to GIFT-64/128 and two GIFT-based AEADs GIFT-COFB and HyENA, and then some differential-linear distinguishers with more rounds are obtained, as summarized in Table 1 where all the results are under the single-key setting.

Differential-linear attacks on two GIFT-based AEADs GIFT-COFB and HyENA. The security concerns of GIFT-COFB and HyENA have attracted considerable attention from many researchers since their publication. There are several attacks on the encryption procedure in message processing phase. In [ZDC+21], Zong et al. gave a key-recovery attack on 15-round GIFT-COFB based on a 9-round linear approximation. Subsequently, Sun et al. [SWW21b] improved this result using the automatic search with the Boolean satisfiability problem (SAT), and gave an attack on 16-round GIFT-COFB with a 10-round linear approximation. Besides, Sun et al. gave a key-recovery attack on 16-round HyENA based on a 10-round linear approximation. With the assistance of our automatic tool, we found 13-round differential-linear distinguishers for GIFT-COFB and HyENA. Then the key-recovery attack is given for 18-round GIFT-COFB, which takes time complexity of $2^{102.06}$ and data complexity of 2^{64} to recover full 128-bit secret key. With regard to 18-round HyENA, we show a key-recovery attack with 2^{119} time complexity and $2^{63.97}$ data complexity. We summarize our attacks and the previous ones against GIFT-COFB and HyENA in Table 2 where all the results are under the single-key setting. Note that for the analysis of the encryption procedure in message processing phase, differential-linear attacks can be launched under the nonce misusing scenario. Moreover, we have analyzed the initialization phase of round-reduced version of GIFT-COFB and HyENA. The attacks on the initialization phase reach 18 rounds for both ciphers, and the details of attack complexities can be found in Table 2.

Evaluation of Security of GIFT-64/128 against Differential-linear Cryptanalysis. Since the publication of GIFT-64/128, there have been plenty of works on their security against differential and linear cryptanalysis. To promote a comprehensive perception of the soundness of GIFT-64/128's security, their actual resistance to the variants of differential or linear cryptanalysis should be evaluated. With our automatic tool, we analyzed the security of GIFT-64/128 against differential-linear attacks. As a result, for 18-round GIFT-64, a key-recovery attack is launched using a 12-round differential-linear approximation. With regard to GIFT-128, a 19-round key-recovery attack is given with a

17-round differential-linear approximation. The details of attack complexities can be found in Table 2. For both ciphers GIFT-64/128, the differential-linear cryptanalysis could not reach the key-recovery attacks with the highest rounds.

As shown in Table 1, with the help of our automatic tool, we found 13-round differential-linear distinguishers for GIFT-COFB and HyENA. For the message processing phase of both AEADs, the distinguishers cover three rounds more than the publicly known results. In virtue of the distinguishers, 18-round key-recovery attacks are given for the message processing phases, as summarized in Table 2, which are better than the previous best ones by two more rounds. Moreover, we have given the attacks on the initialization phases of 18-round GIFT-COFB and HyENA respectively, which facilitates our understanding of their security in different phases. For GIFT-64, as shown in Table 1, a 12-round differential-linear distinguisher is found which has the same rounds as the linear one in [SWW21a] but one less round than the differential one in [CZD19]. As regards GIFT-128, a 17-round differential-linear distinguisher is found, which has four or two fewer rounds with the differential [JZZD20a] or linear [SWW21b] ones respectively. Then, 18-round and 19-round key-recovery attacks are given for GIFT-64 and GIFT-128 respectively, which could not reach the same rounds with the best attacks obtained by differential cryptanalysis in [CZD19] and [ZDC+21] respectively, same to the linear case. For the details of attacks, please refer to Table 2.

Table 1: Summary of distinguishers on GIFT-64/128, GIFT-COFB and HyENA. For GIFT-64/128, the attacks target on the encryption phase (Enc. for short). For GIFT-COFB and HyENA, the initialization phase (Init. P.) and message processing phase (Msg. P.) are analyzed. For different types of distinguishers, Diff. denotes for differential, Lin. for linear and DL for differential-linear. PR denotes the probability of differential distinguisher and SC denotes the squared correlation of linear and differential-linear distinguishers.

Cipher	Target	Rounds	Type	PR (SC)	Ref.
GIFT-COFB	Msg. P.	9	Lin.	2^{-58}	[ZDC+21]
		10	Lin.	$2^{-57.68}$	[SWW21b]
		13	DL	$2^{-57.56}$	Sect. 4.1
	Init. P.	13	DL	$2^{-55.56}$	Sect. 4.2
HyENA	Msg. P.	10	Lin.	$2^{-55.36}$	[SWW21b]
		13	DL	$2^{-59.02}$	Sect. 5.1
	Init. P.	13	DL	$2^{-59.02}$	Sect. 5.1
GIFT-64	Enc.	9	Diff.	$2^{-44.415}$	[BPP+17]
		12	Diff.	2^{-60}	[ZDY19]
		12	Diff.	$2^{-56.57}$	[CZD19]
		13	Diff.	$2^{-61.31}$	[CZD19]
		9	Lin.	$2^{-49.997}$	[BPP+17]
		12	Lin.	$2^{-61.61}$	[SWW21a]
		12	DL	$2^{-57.22}$	Sect. 6.1
GIFT-128	Enc.	9	Diff.	$2^{-46.99}$	[BPP+17]
		18	Diff.	2^{-109}	[ZDY19]
		20	Diff.	$2^{-120.245}$	[JZZD20b]
		21	Diff.	$2^{-126.415}$	[JZZD20a]
		20	Diff.	$2^{-121.81}$	[ZDC+21]
		15	Lin.	2^{-109}	[ZDC+21]
		19	Lin.	$2^{-117.43}$	[SWW21b]
		19	Lin.	$2^{-123.11}$	[SWW22]
		17	DL	$2^{-117.56}$	Sect. 6.2

Refer to `https://gitfront.io/r/user-9335734/A33hSkkf6eEa/DL-GIFT/` for the full version of this paper with the supplementary material where the codes and details of attacks on GIFT-COFB, HyENA and GIFT-64/128 are provided.

1.2 Organization of This Paper

The rest of paper is organized as follows. In Sect. 2, we introduce specifications of GIFT family and GIFT-COFB, HyENA, and recall the MILP-based automatic search method and differential-linear cryptanalysis. In Sect. 3, we present the automatic tool i.e. MILP model to search differential-linear approximations for the S-box-based ciphers. The details of differential-linear attacks on GIFT-COFB, HyENA and GIFT-64/128 are shown in Sect. 4, Sect. 5 and Sect. 6 respectively. Finally, we conclude this paper in Sect. 7.

Table 2: Summary of attacks on GIFT-64/128, GIFT-COFB and HyENA. For GIFT-64/128, the target of attacks is the encryption phase (Enc. for short). For two AEADs GIFT-COFB and HyENA, we consider the attacks on the initialization phase (Init. P.) and message processing phase (Msg. P.). For different types of attacks, Diff denotes for differential attacks, Lin. for linear attacks and DL for differential-linear attacks.

Cipher	Target	Rounds	Type	Time	Data	Memory	Ref.
GIFT-COFB	Msg. P.	15	Lin.	$2^{90.70}$	$2^{62.00}$	2^{96}	[ZDC+21]
		16	Lin.	$2^{122.80}$	$2^{62.10}$	2^{47}	[SWW21b]
		18 †	DL	$2^{102.06}$	2^{64}	negligible	Sect. 4.1
	Init. P.	18	DL	$2^{97.88}$	$2^{62.41}$	negligible	Sect. 4.2
HyENA	Msg. P.	16	Lin.	$2^{122.00}$	$2^{61.51}$	2^{52}	[SWW21b]
		18 †	DL	2^{119}	$2^{63.97}$	negligible	Sect. 5.2
	Init. P.	18	DL	2^{119}	$2^{63.97}$	negligible	Sect. 5.2
GIFT-64	Enc.	19	Diff.	2^{112}	2^{63}	2^{80}	[ZDY19]
		20	Diff.	$2^{101.68}$	2^{64}	2^{96}	[CZD19]
		21	Diff.	$2^{107.61}$	2^{64}	2^{96}	[CZD19]
		19	Lin.	$2^{127.11}$	$2^{62.96}$	2^{60}	[SWW21a]
		18	DL	$2^{124.61}$	$2^{61.57}$	negligible	Sect. 6.1
GIFT-128	Enc.	22	Diff.	2^{114}	2^{114}	2^{53}	[ZDY19]
		26	Diff.	$2^{123.245}$	$2^{123.245}$	2^{109}	[JZZD20b]
		27	Diff.	$2^{124.83}$	$2^{123.53}$	2^{80}	[ZDC+21]
		22	Lin.	$2^{117.00}$	$2^{117.00}$	2^{78}	[ZDC+21]
		24	Lin.	$2^{124.45}$	$2^{122.55}$	2^{105}	[SWW21b]
		25	Lin.	$2^{126.77}$	$2^{124.75}$	2^{96}	[SWW22]
		19	DL	$2^{121.53}$	$2^{122.51}$	negligible	Sect. 6.2

† Launched under the nonce misusing scenario.

2 Preliminaries

In this section, we first introduce the specifications of GIFT, GIFT-COFB and HyENA. Then we recall the MILP-based automatic search method and differential-linear cryptanalysis.

2.1 Description of GIFT

GIFT, proposed by Banik et al. [BPP+17] at CHES 2017, has two versions, namely GIFT-64 and GIFT-128. Both of them have the same key length of 128 bits, while the block sizes are 64 and 128 respectively. Here we mainly introduce the description of GIFT-128, and the similar structure to GIFT-64. For more details, please refer to [BPP+17].

GIFT-128 follows an SPN structure with 40 rounds. The round function has three steps: *SubCells*, *PermBits* and *AddRoundkey* which are illustrated as follows.

*SubCells***.** The Sbox of GIFT-128, denoted by GS, can by found in the full paper. In each round, the state is updated by applying 32 GS operations in parallel to every nibble.

*PermBits***.** Then update the cipher state by a linear transformation $P_{128}(\cdot)$ as $b_{P_{128}(i)} \leftarrow b_i$, $i = 0, 1, \cdots, 127$. Refer to the full paper for details.

*AddRoundKey***.** A 64-bit round key is viewed as two 32-bit words. In another way, $RK = U||V = u_{31} \cdots u_0||v_{31} \cdots v_0$. Then half of the internal state bits are XORed with RK as the following shows: $b_{4i+2} \leftarrow b_{4i+2} \oplus u_i, b_{4i+1} \leftarrow b_{4i+1} \oplus v_i$, $\forall i \in \{0, \ldots, 31\}$.

GIFT-COFB and HyENA. The specification of GIFT-COFB and HyENA is in the full paper.

Here we summarize the notations used in our attacks as Table 3.

Table 3: The notations of GIFT

X_i	:	The input state of i-th round, and $X_1 = P$
X_i^S	:	The state after *Subcells* transformation of i-th round
X_i^P	:	The state after *PermBits* transformation of i-th round
$X_i^{S,K}$	:	$PermBits^{-1}(X_{i+1})$
ΔX	:	The difference of state X
$X[i]$	:	The i-th bit of state X, and $X[0]$ is the LSB of X
RK_i	:	The round key of i-th round
RK'_i	:	$PermBits^{-1}(RK_i)$
$RK[i]$	:	The i-th bit of round key, and the same to $RK'[i]$

2.2 Automatic Search Methods for Differential and Linear Trails

The automatic search method will be recalled in this section. Mouha et al. [MWGP11] showed that the problem of searching for the minimum number of active S-boxes can be modeled with mixed integer linear programming (MILP), which is effective for evaluating word-oriented ciphers. To apply MILP to bit-oriented ciphers, Sun et al. [SHW+14b] developed a method to model all possible differential propagation bit by bit for the S-box. In the following, we briefly review the method in [SHW+14b]. Owing to the similarity of the modeling procedure between searching for differential and linear trails, we omit the case of linear cryptanalysis for convenience narration.

Definition 1. Suppose a n-bit differential characteristic state $\Delta = (\Delta_0, \Delta_1, \cdots, \Delta_{n-1})$. We define the vector $x = (x_0, x_1, \cdots, x_{n-1})$ to mark the active or inactive bit positions as follows:

$$x_i = \begin{cases} 0, & \text{if } \Delta_i = 0, \\ 1, & \text{if } \Delta_i = 1. \end{cases} \tag{1}$$

Constraints of S-box. Suppose the two vectors $(x_0, x_1, \cdots, x_{\omega-1})$ and $(y_0, y_1, \cdots, y_{\nu-1})$ are the input and output bit differences of some $\omega \times \nu$ S-box S_t. Let the bit variable A_t denote the activity of this S-box. That is to say, $A_t = 1$ if S_t is active, and $A_t = 0$ otherwise. The following constraints can be used to ensure that the non-zero input difference of the S-box must activate it:

$$\begin{cases} A_t - x_k \geq 0, \ k = 0, \ldots, \omega - 1, \\ -A_t + \sum_{j=0}^{\omega-1} x_j \geq 0. \end{cases}$$

To describe the differential propagation with probabilities, we introduce a vector $(x_0, \cdots, x_{\omega-1}, y_0, \cdots, y_{\nu-1}, p_t, q_t) \in \mathbb{R}^{\omega+\nu+2}$ and then get a finite set of discrete points that just includes all the possible differential propagations and their corresponding probabilities of the S-box. And the above set can be represented by the inequalities called the H-representation of the S-box S_t :

$$\begin{cases} \alpha_{0,0}x_0 + \ldots + \alpha_{0,w-1}x_{w-1} + \ldots + \beta_{0,v-1}y_{v-1} + \gamma_{0,1}p_t + \gamma_{0,2}q_t + \delta_0 \geq 0, \\ \cdots\cdots, \\ \alpha_{n,0}x_0 + \ldots + \alpha_{n,w-1}x_{w-1} + \ldots + \beta_{n,v-1}y_{v-1} + \gamma_{n,1}p_t + \gamma_{n,2}q_t + \delta_n \geq 0. \end{cases}$$

We can utilize the existing algorithm of SageMath to derive inequalities to represent the propagation of differential or linear masks of the S-box, and then reduce their number by greedy algorithm given in [SHW$^+$14a].

Objective function of differential propagation model. The objective function should be a linear function of variables and can be the minimum number of active S-boxes $\sum A_t$ or the highest probability of differential trails $\sum p_t + \sum q_t$ for the cipher.

2.3 Differential-Linear Cryptanalysis

In the following, we recall the common framework of differential-linear approximation and the success probability of a key-recovery attack in the differential-linear context.

In practice, the assumption of independence between two subciphers might lead to the wrong estimation of the correlation of differential-linear approximation. Usually, one can get some evidence of this independence assumption by computing experimentally the correlation of differential-linear approximation over round-reduced cipher. To obtain a more accurate estimation of the differential-linear approximation, the target cipher is divided into three parts E_d, E_m and E_l such that $E = E_l \circ E_m \circ E_d$ like in recent works [Leu16, BDKW19, BLT20]. The overall framework of differential-linear approximation is illustrated in Figure 1. Bar-On et al. [BDKW19] introduced a theoretical method called DLCT to characterize the property of middle part E_m. However, it is still a question about how to expand the DLCT to cover more rounds. Subsequently, Beierle et al. [BLT20] presented several improvements in differential-linear attacks for ARX ciphers. In their work, the correlation of middle part E_m was experimentally evaluated. Assume that a differential $\Delta_{in} \xrightarrow{p} \Delta_m$ for E_d holds with probability $\Pr[E_d(P) \oplus E_d(P \oplus \Delta_{in}) = \Delta_m] = p$, and that a linear approximation $\Gamma_m \xrightarrow{q} \Gamma_{out}$ for E_l holds with probability $\Pr[\Gamma_m \cdot Y = \Gamma_{out} \cdot E_l(Y)] = \frac{1}{2}(1+q)$, and the approximation for middle part E_m holds with probability $\Pr[\Gamma_m \cdot E_m(X) = \Gamma_m \cdot E_m(X \oplus \Delta_m)] = \frac{1}{2}(1+r)$ (or with correlation r), where $\cdot$ denotes the inner product between two vectors. Under the assumption of independence between subciphers, the probability of differential-linear approximation can be simply estimated using Piling-up Lemma, $\Pr[\Gamma_{out} \cdot E(P) = \Gamma_{out} \cdot E(P \oplus \Delta_{in})] = \frac{1}{2}(1+prq^2)$. Therefore, one can distinguish the cipher E from a random permutation using $N = O(p^{-2}r^{-2}q^{-4})$ chosen plaintext pairs $(P, P \oplus \Delta_{in})$.

Success Probability. In [BLN17], Blondeau et al. gave the success probability of a key-recovery attack in the differential-linear context by adapting the one of linear cryptanalysis in [Sel08],

$$P_S = \Phi(2\sqrt{N}|p_{dl} - \frac{1}{2}| - \Phi^{-1}(1-2^{-a})), \tag{2}$$

where Φ is the cumulative distribution function of the standard normal distribution, p_{dl} is the probability of differential-linear distinguisher, N is the number of chosen plaintext pairs and a is the advantage of attack as defined in [Sel08].

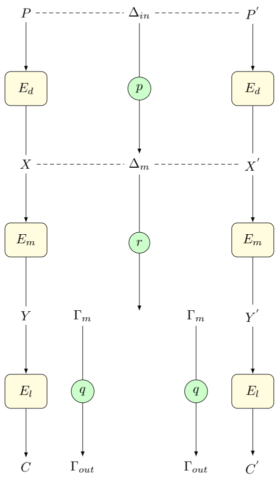

Figure 1: The framework of differential-linear approximation

3 Automatic Tool of DL Approximation for S-box-Based Ciphers

In this section, we first present the automatic tool i.e. a MILP model to search differential-linear (DL) approximations for the overall framework shown in Figure 1. For efficiently computing the correlation of middle part E_m in the MILP model, we show how to derive the propagation of correlation of differential-linear approximations for S-boxes. At last, a theoretical estimation of the correlation of DL approximation is given for `GIFT-128`.

3.1 MILP Model of Searching DL Approximations

The R-round cipher E is divided into three parts r_d-round E_d, r_m-round E_m and r_l-round E_l, namely, $E = E_l \circ E_m \circ E_d$ and $R = r_d + r_m + r_l$. Let Δ_m and Γ_m be the input difference and output linear mask of E_m respectively.

Modeling the middle part E_m. As reviewed in Sect. 2.2, there have been automatic tools with the MILP model to search differential and linear trails for the S-box-based ciphers. So, the crucial point is how to model for differential-linear approximations of E_m with the MILP method. Similarly, for the differential-linear part E_m (treated as a whole part), a super table can be defined, which is actually *Differential Linear Connectivity Table* (DLCT). Since most entries of DLCT for E_m have small correlations and may not lead to good solutions, we can manually exclude them by some constraints, e.g., the partial DLCT with single-bit active input difference and output linear mask. Once the correlations of restricted input difference and output linear mask of E_m are computed, we store them in a table as the partial DLCT of E_m, denoted as $C_m[\cdot]$.

The remaining and central question is how to encode the partial DLCT with correlation into the MILP model. Note that the objective function is required to be linear in the MILP model. For encoding the correlation of partial DLCT, we introduce the auxiliary variables. Precisely, for the input difference δ_m and output linear mask γ_m of E_m, an auxiliary variable z_{δ_m,γ_m} is introduced. When $\Delta_m = \delta_m$ and $\Gamma_m = \gamma_m$, the auxiliary variable z_{δ_m,γ_m} equals to one; otherwise zero. So, an auxiliary variable is set for choosing the specific pattern of difference and linear mask in E_m. To model the rule of choosing the pattern of E_m in MILP, we derive Theorem 1 to express the pattern-choosing rule with linear inequalities.

Theorem 1. *Let a vectorial variable* $(x_0, x_1, \cdots, x_{n-1}) \in \{0,1\}^n \subset \mathbb{Z}^n$ *and a variable* $z \in \{0,1\} \subset \mathbb{Z}$*, constants* $(\alpha_0, \alpha_1, \cdots, \alpha_{n-1}) \in \{0,1\}^n$ *and* $\beta \in \{0,1\}$*. Then the pattern-choosing rule that* $(x_0, x_1, \cdots, x_{n-1}) = (\alpha_0, \alpha_1, \cdots, \alpha_{n-1})$ *if and only if* $z = \beta$ *can be equivalently described by the following two inequalities:*

$$\sum_{i=0}^{n-1}(-1)^{\alpha_i}x_i + (-1)^{\beta+1}z + \sum_{i=0}^{n-1}\alpha_i - \beta \geq 0,$$
$$\sum_{i=0}^{n-1}(-1)^{\alpha_i+1}x_i + m(-1)^{\beta}z - \sum_{i=0}^{n-1}\alpha_i + m\beta \geq 0,$$

where $m \geq n$*.*

Proof. Note that the expression $\alpha + (-1)^\alpha x$ equals to 0 when $x = \alpha$, and equals to 1 when $x \neq \alpha$, for $x, \alpha \in \{0,1\}$.

For the condition that $(x_0, x_1, \cdots, x_{n-1}) = (\alpha_0, \alpha_1, \cdots, \alpha_{n-1})$, the upper inequality excludes the possibility that $z \neq \beta$ and the lower inequality always holds. For the condition that $(x_0, x_1, \cdots, x_{n-1}) \neq (\alpha_0, \alpha_1, \cdots, \alpha_{n-1})$, the lower inequality excludes the possibility that $z = \beta$ and the upper inequality always holds.

Therefore, the two cases $(x_0, x_1, \cdots, x_{n-1}) = (\alpha_0, \alpha_1, \cdots, \alpha_{n-1})$, $z \neq \beta$ and $(x_0, x_1, \cdots, x_{n-1}) \neq (\alpha_0, \alpha_1, \cdots, \alpha_{n-1})$, $z = \beta$ can not make the system of the above two inequalities satisfied. While the other two cases $(x_0, x_1, \cdots, x_{n-1}) = (\alpha_0, \alpha_1, \cdots, \alpha_{n-1})$, $z = \beta$ and $(x_0, x_1, \cdots, x_{n-1}) \neq (\alpha_0, \alpha_1, \cdots, \alpha_{n-1})$, $z \neq \beta$ always satisfy the system. □

Theorem 1 is actually the extension of the one in [SHW$^+$14b]. According to Theorem 1, we can use two inequalities to describe the pattern-choosing rule that $(\Delta_m, \Gamma_m) = (\delta_m, \gamma_m)$ if and only if $z_{\delta_m,\gamma_m} = 1$. So, the correlation of E_m can be expressed as $\sum z_{\delta_m,\gamma_m} C_m[\delta_m, \gamma_m]$ which is a linear function and can be used in MILP models.

Modeling the parts E_d and E_l. Here, we briefly describe how to construct MILP models to search linear and differential trails of `GIFT-128`. The details for modeling the linear part of E_l are as follows. For the S-box of `GIFT-128`, since there are 3 possible correlations, i.e., 1, 2^{-1}, 2^{-2}, we add two extra bits (q_0, q_1) to encode the correlation of the linear mask propagation. Therefore, a vector $(x_0, \cdots, x_3, y_0, \cdots, y_3, q_0, q_1) \in \mathbb{R}^{10}$ can describe a linear mask pattern with correlation for the S-box. Then by SageMath, 454 inequalities are derived through computing the H-Representation of the convex hull, and the number of inequalities is reduced to 20 by greedy algorithm in [SHW$^+$14a]. Since the $PermBits(\cdot)$ transform is a simple permutation on a 128-bit state, there is no need to introduce new inequalities. Besides, we can ignore the *AddRoundKey* transform in the linear trail (actually in the differential-linear context). The correlation of the linear trail through E_l is expressed as $\sum(q_0 + 2q_1)$.

With regard to the differential trails, the modeling process is similar to the aforementioned. In [ZDY19], they presented the MILP-based automatic method to search differential trails for `GIFT-128`. For the differential part of E_d, we just adopt their method to model the differential patterns with their probabilities for the S-box of `GIFT-128`. Refer to [ZDY19] for details. The probability of differential trail through E_d is denoted as $\sum(3p_0 + 2p_1 + 1.415p_2)$.

As a result, we integrate the three parts into a whole MILP model to search for differential-linear approximations. The objective function is minimization of the formula $\sum(3p_0 + 2p_1 + 1.415p_2) + \sum z_{\delta_m,\gamma_m} C_m[\delta_m, \gamma_m] + \sum(2q_0 + 4q_1)$, which denotes the total correlation of differential-linear approximation through the cipher $E = E_l \circ E_m \circ E_d$.

Single-bit input difference and output linear mask. Let b denote the state size of the block cipher. There are b possibilities of all the single-bit input difference, and the same for single-bit output mask. The single-bit input difference δ_m is determined by the position of the active bit in δ_m, same for output mask γ_m. So, an auxiliary variable z_{δ_m,γ_m} depends on two bits, i.e. one bit for input difference and one bit for output mask. In the simplest case, the pattern-choosing rule can be interpreted as the AND operation, i.e., the auxiliary variable z equals one if and only if the bit x_0 of input difference and the bit x_1 of output mask are both one. According to Theorem 1, we can use two inequalities to describe the pattern-choosing rule, that is $z + 1 \geq x_0 + x_1$ and $x_0 + x_1 \geq 2z$. As a result, there are in total $b \times b$ auxiliary variables z_{δ_m,γ_m} for choosing the pattern of output difference and linear mask of E_m.

For the more general cases, the method can be applied, but there will be too many auxiliary variables so the MILP model will be very time-consuming.

3.2 Propagation of Correlation of DL Approximations for S-box

In this subsection, we show how to theoretically and efficiently estimate the correlation of DL approximation for the S-boxes of ciphers, given their DDTs or ANFs.

Calculating the correlation of DL approximations for GS from its DDT. To obtain the propagation rules of DL approximation for S-boxes, we first recall Observation 1 in [LSL21]. For the case of differential-linear approximation of S-boxes, let $x = (x_{n-1}, x_{n-2}, \cdots, x_0)$ be the input of a cipher's S-box $S : \mathbb{F}_2^n \to \mathbb{F}_2^n$ and $y = (y_{n-1}, y_{n-2}, \cdots, y_0) = S(x)$ be the output. Let Δx denote the input difference between x and another input x', i.e., $\Delta x = x \oplus x'$ with correlation $c_i = Cor[\Delta x_i] = 2\Pr[\Delta x_i = 0] - 1$, and $\Delta y = y \oplus y'$ the output difference where $y' = S(x')$. Then the probability/correlation of output difference $\Delta y_i = 0$ can be determined by the following formula

$$\begin{aligned}\Pr[\Delta y_i = 0] &= \sum_{\Delta x \in \mathbb{F}_2^n} \Pr[x \oplus x' = \Delta x] \Pr[\Delta y_i = 0 \mid x \oplus x' = \Delta x] \\ &= \sum_{\Delta x \in \mathbb{F}_2^n} (\prod_{j=0}^{n-1} \frac{1 + (-1)^{\Delta x_j} c_j}{2}) \times \frac{\#\{x \mid (S(x \oplus \Delta x) \oplus S(x))[i] = 0\}}{2^n},\end{aligned} \tag{3}$$

and $Cor[\Delta y_i] = 2\Pr[\Delta y_i = 0] - 1$.

Algorithm 1: Deriving Propagation of DL approximation for S-box from DDT

```
Input: S-box's DDT and variables $c_i$ for $i = 0, 1, \cdots, n-1$.
Output: Propagation rule of DL approximation's correlation for S-box.
Initialize an $n$-dimension array $p[\cdot]$ with all zeros.
for $\Delta x \in \mathbb{F}_2^n$ do
    $\Pr[\Delta x] = \prod_{i=0}^{n-1} \frac{1+(-1)^{\Delta x_i} c_i}{2}$
    for $\Delta y \in \mathbb{F}_2^n$ do
        for $i = 0, 1, \cdots, n-1$ do
            if $\Delta y_i = 0$ then
                $p[i] \leftarrow p[i] + 2^{-n}\mathrm{DDT}[\Delta x][\Delta y] \times \Pr[\Delta x]$
            end
        end
    end
end
return $2p[i] - 1$ for $i = 0, 1, \cdots, n-1$
```

Next, from the view of DDT of S-box, we present a way, i.e., Algorithm 1, to derive the propagation of correlation of DL approximation for S-boxes. This works as follows: for a fixed row of DDT, sum the entries of DDT first column by column which is multiplied by a row-dependent probability, then sum the results for all rows. Algorithm 1 gives an interpretation of the propagation rule of DL approximation from the perspective of differential. Note that we can focus on and sum over the non-zero entries of DDT in Algorithm 1. From the view of DDT, the propagation of DL approximation of S-box is a row-weighted sum of all non-zero entries of DDT. So the propagation of DL approximation takes all possibilities of input/output difference into account with the distribution of input difference under some independence assumption.

We have implemented Algorithm 1 by the symbolic programming in SageMath. Refer to the full paper for the details of codes. The implementation in SageMath code can also be easily used to analyze other ciphers' S-boxes, such as PRESENT and SKINNY, see the full paper for the details.

Calculating the correlation of DL approximations for GS from its ANF. Now we discuss the DL approximation of S-box from another view, i.e., an algebraic view as proposed in [LLL21]. By replacing $x\Delta_{in}$ with a variable vector for input difference and adding its distribution into D in Algorithm 3 of [LLL21], we derive the propagation of the DL approximation of the S-box for an arbitrary output linear mask, as depicted in Algorithm 2.

```
Algorithm 2: Deriving Propagation of DL approximation for S-box from ANF
Input: The ANF $F$ of the S-box, the correlation $\{c_i\}$ of input difference, and an
       output mask $\lambda$.
Output: Propagation rule of DL approximation's correlation for S-box.
Initialize a variable vector $\Delta x$ for input differential, and initialize a probability
  distribution set $D = \{\Pr[x_i = 0] = \frac{1}{2}\} \cup \{\Pr[\Delta x_i = 0] = \frac{1+c_i}{2}\}$.
Compute the output difference $\Delta y_\lambda = \lambda \cdot (F(x) \oplus F(x \oplus \Delta x))$ which is Boolean
  function of $2n$ variables $x_i$'s and $\Delta x_i$'s for $i = 0, 1, \cdots, n-1$.
With the assumption that $x_i$'s and $\Delta x_i$'s for $i = 0, 1, \cdots, n-1$ are independent,
  compute the probability of $\Delta y_\lambda$ according to Equation (3.2) in [LLL21], i.e.,
  $\Pr[\Delta y_\lambda = 0] = \sum_{a \in \{(x,\Delta x) | \Delta y_\lambda = 0\}} 2^{-n} \prod_{i=0}^{n-1} (\frac{1+(-1)^{a_{n+i}} c_i}{2})$.
return $2\Pr[\Delta y_\lambda = 0] - 1$.
```

By performing either Algorithm 1 or Algorithm 2, we get the following proposition to describe the propagation of correlation of DL approximation for the S-box of GIFT.

Proposition 1. *Let $x = (x_3, x_2, x_1, x_0)$ be input of S-box of GIFT GS, and $\Delta x = (\Delta x_3, \Delta x_2, \Delta x_1, \Delta x_0)$ be input difference with correlation $Cor[\Delta x_i] = c_i$, i.e., probability $\Pr[\Delta x_i = 0] = \frac{1+c_i}{2}$, for $i \in \{0, 1, 2, 3\}$. The corresponding output difference of GS is denoted by $\Delta y = (\Delta y_3, \Delta y_2, \Delta y_1, \Delta y_0)$, i.e., $y = GS(x)$ and $\Delta y = GS(x) \oplus GS(x \oplus \Delta x)$. Assuming the bits of x and Δx are independent, we have*

$$\begin{aligned}
Cor[\Delta y_0] &= \frac{1}{4}(c_0+1)(c_1+1)c_2c_3, \\
Cor[\Delta y_1] &= \frac{1}{4}(c_0+1)(c_1+c_2)c_3, \\
Cor[\Delta y_2] &= \frac{1}{16}(c_0c_1c_2 + c_0c_1 + c_0c_2 + 4c_1c_2 + c_0)(c_3+1), \\
Cor[\Delta y_3] &= \frac{1}{16}(c_0c_1c_2 + c_0c_1 + c_1c_2 + 4c_0 + c_1)(c_3+1).
\end{aligned} \tag{4}$$

Proposition 1 considers the propagation of differences in single bits, and we refer to the full paper for the DL approximation of the S-box with an arbitrary output linear mask.

Independence assumption of Proposition 1. The assumption of the independence of input bits always holds, since the `GIFT` round function is a bijection. The independence assumption of the four bits of input difference is reasonable for GS because the four bits input to one S-box originate from different S-boxes from the previous round due to the property of bit permutation in the linear layer. Our experiments show that the independence between the four bits of the input x and the four bits of the input difference Δx are also reasonable. More exactly, we have verified that $\Pr[(x, \Delta x) = a] = 2^{-n} \prod_{i=0}^{n-1} (\frac{1+(-1)^{a_{n+i}} c_i}{2})$ holds with a probability greater than 0.98, within the allowed error range of 10%. In the experiments, we set x and Δx to the bits and difference bits input to the same S-box of the sixth round of `GIFT`, and the input difference of the first round to a random difference with Hamming weight up to three, and verify the equation for all possible a and repeat it for hundreds of times.

3.3 Estimation of Correlation of DL Approximation for `GIFT-128`

Here, we give an example of `GIFT-128` to demonstrate the theoretical estimation of DL approximation's correlation by using Proposition 1.

In Figure 2, for r-round `GIFT-128`, blue symbol x denotes the logarithm of maximum correlation of single-bit input difference and output mask which is theoretically estimated by Proposition 1 and black circle the one estimated by the sampling experiment, and red line the error percentage which is defined as the absolute value of the difference between the theoretical value and the experimental value, divided by the experimental value. Since at most 2^{34} random plaintext pairs are used for each of 2^3 random keys, a correlation of about $|Cor| > c \cdot 2^{-17} = 2^{-13.5}$ can be detected (where $c \approx \sqrt{128}$ for reasonable estimation error). As shown in Figure 2, the theoretical estimations of correlation of DL approximation match the experimental results in the first eight rounds, and the error percentage remains within 55%. For the trend of correlation with an increasing number of rounds, the correlation of DL approximation decreases for `GIFT-128`, especially decreasing sharply after eight rounds.

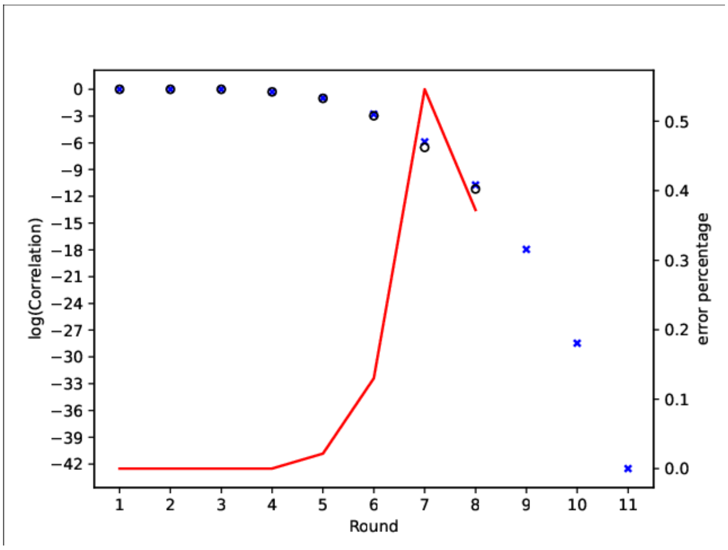

Figure 2: Estimation of maximum correlation for r-round `GIFT-128`, where x denotes the theoretical one, black circle the experimental one, and red line the error percentage.

4 Differential-Linear Attacks on GIFT-COFB

In this section, we present our differential-linear attacks on GIFT-COFB, including the attacks on message processing phase and initialization phase.

4.1 Attack on Message Processing Phase

We first present the procedure of automatically searching differential-linear distinguishers with the assistance of the MILP model. Then the key-recovery attack on GIFT-COFB is given based on the new distinguisher.

Before showing the details of the analysis, let us take a look at the restriction of active bits in attacks on the message processing phase of GIFT-COFB. For the attack on message processing phase which is illustrated in the figure of the full paper. From the plaintext-ciphertext pair of GIFT-COFB $(M[1]||M[2], C[1]||C[2])$, we can get the input-output pair of the cipher GIFT-128, and the input is $G(Y[a]) \oplus M[1] \oplus (2^a 3^i L||0^{n/2})$, the corresponding output is $M[2] \oplus C[2]$, where $Y[a] = M[1] \oplus C[1]$. Under the nonce misusing scenario, another input-output pair can be chosen for GIFT-128. Since L is unknown (depending on nonce and secret key), we can not get the value from the most significant 64 bits of the input for GIFT-128 back to $(G(Y[a]) \oplus M[1])[64-127]$. So, the data structure in the key-recovery attacks should not involve the most significant 64 bits of GIFT-128's input.

As stated in [SWW21b], *Given the GIFT-128 achieves full diffusion after four rounds, we conjecture the maximum number of rounds annexed before the linear distinguisher in the attack on GIFT-COFB is three.* Similarly, the maximum number of rounds extended before the differential-linear approximation is assumed as three. In [SWW21b], they introduced extra variables and Boolean expressions in their model to satisfy the restriction that there are no active bits in the most significant 64 bits of the input for GIFT-128.

By exploiting the structure property of GIFT-128, we have found a phenomenon of the differential propagation of 3-round GIFT-128. The phenomenon is summarized in Proposition 2, which reveals that the most significant 64 bits of GIFT-128's input always go to another fixed set of 64 bits after three rounds. According to Proposition 2, the restriction of active bits for key-recovery attacks can be directly converted into the restriction for distinguisher's inputs.

Proposition 2. *Let ΔP be the input difference of plaintext, and ΔX_4 be the input difference of the 4-th round, i.e., the output difference of the 3-rd round. If $Index(\Delta X_4) \subseteq S$, then $Index(\Delta P) \subseteq \{0, 1, \cdots, 63\}$ after three rounds backward; vice versa, where the $Index(\cdot)$ function returns the indices on which the value is non-zero, and $S = \{4j_0, 4j_0+2 | j_0 \in \{0, 1, \cdots, 7, 16, 17, \cdots, 23\}\} \cup \{4j_1+1, 4j_1+3 | j_1 \in \{8, 9, \cdots, 15, 24, 25, \cdots, 31\}\}$.*

Proof. Due to the invertibility of GS, the input difference at a single S-box is zero if and only if the output difference is zero. Besides, the transformation *SubCells*, which applies 32 parallel S-boxes, does not change the position of bits. Therefore, the function $Index(\cdot)$ remains unchanged through the transformation *SubCells* in terms of the S-boxes.

Since the set of bit positions $S_1 = \{0, 1, \cdots, 63\}$ corresponds to bit positions of the $\{0, 1, \cdots, 15\}$ S-boxes, then $Index(\Delta P) \subseteq S_1$ if and only if $Index(\Delta X_1^S) \subseteq S_1$. Because the transformation *PermBits* maps the set S_1 to the set of bit positions $S_2 = \{0, 1, \cdots, 11\} \cup \{32, 33, \cdots, 43\} \cup \{64, 65, \cdots, 75\} \cup \{96, 97, \ldots, 107\}$, so $Index(\Delta X_1^S) \subseteq S_1$ if and only if $Index(\Delta X_1^P) \subseteq S_2$. We can ignore the transformation *AddRoundKey*, because it does not affect the propagation of differences.

Since the set S_2 corresponds to bit positions of the $\{0, 1, 2, 3, 8, 9, 10, 11, 16, 17, 18, 19, 24, 25, 26, 27\}$ S-boxes, then $Index(\Delta X_1^P) \subseteq S_2$ if and only if $Index(\Delta X_2^S) \subseteq S_2$. Because the transformation *PermBits* maps the set S_2 to the set of bit positions $S_3 = \cup_{i=0}^{15} \{8i + j | j = 0, 1, 2, 3\}$, so $Index(\Delta X_2^S) \subseteq S_2$ if and only if $Index(\Delta X_2^P) \subseteq S_3$.

Since the set S_3 corresponds to bit positions of $\{2i | i = 0, 1, \cdots, 15\}$ S-boxes, then $Index(\Delta X_2^P) \subseteq S_3$ if and only if $Index(\Delta X_3^S) \subseteq S_3$. Because the transformation $PermBits$ maps the set S_3 to the set of bit positions $S_4 = \{4j_0, 4j_0 + 2 | j_0 \in \{0, 1, \cdots, 7, 16, 17, \cdots, 23\}\} \cup \{4j_1 + 1, 4j_1 + 3 | j_1 \in \{8, 9, \cdots, 15, 24, 25, \cdots, 31\}\}$, so $Index$ $(\Delta X_3^S) \subseteq S_3$ if and only if $Index(\Delta X_3^P) \subseteq S_4$, the same for $Index(\Delta X_4)$.

This completes the proof.

□

Therefore, the specialized MILP model to search DL approximations for `GIFT-COFB` should be set such that there is no active bit in $\{0, 1, \cdots, 127\} \setminus S$, where S is defined in Proposition 2. From Proposition 2, the DL distinguishers returned by the specialized MILP model will always satisfy the restriction in key-recovery attack.

Table 4: The attack on message processing phase of 18-round `GIFT-COFB`

ΔP	---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- **** **** **** **** **** **** **** **** **** **** **** **** **** **** **** ****
ΔX_1^S	---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- -**- --** 1--* *1-- -**- --** *--* **-- -**- --** *--* **-- -**- --** *--* **--
ΔX_1^P	---- ---- ---- ---- **** **** **** **** ---- ---- ---- ---- 11** **** **** **** ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ----
RK_1	-- -- -- -- 11 11 11 11 -- -- -- -- 11 11 11 11 -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
ΔX_2	---- ---- ---- ---- **** **** **** **** ---- ---- ---- ---- 11** **** **** **** ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ----
ΔX_2^S	---- ---- ---- ---- -**- --** 1--* 1*-- ---- ---- ---- ---- -1-- --*- ---* 1--- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ----
ΔX_2^P	---- 1*** ---- 11** ---- ---- ---- ---- ---- 1*** ----
RK_2	-- 11 -- 11 -- -- -- -- -- 11 --
ΔX_3	---- 1*** ---- 11** ---- ---- ---- ---- ---- 1*** ----
ΔX_3^S	---- ---1 ---- -1-- ---- ---- ---- ---- ---- ---1 ----
ΔX_3^P	---- ---- ---- ---- ---- ---- ---- ---- -1-1 ---- ---1 ----
RK_3	-- --
Δ_{in}	---- ---- ---- ---- ---- ---- ---- ---- -1-1 ---- ---1 ----
Γ_{out}	---- ---- ---- ---- ---- --1- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- 1--- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- -1-- ---- ----
X_{17}^S	---- ---- ---- ---- ---- ●●●● ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ●1●1 ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ●11● ---- ----
RK'_{17}	-- -- -- -- -- 11 -- -- -- -- -- -- -- -- -- -- -- -1 -- -- -- -- -- -- -- -- -- -- -- -- -- --
$X_{17}^{S,K}$	---- ---- ---- ---- ---- ●●●● ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ●1●1 ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ●11● ---- ----
X_{18}	---- --●- ---- ---- --●- ---- ---- --1- ---- ---● ---- ---- ---1 ---- ---- ---● ---- ●--- ---- ---- ●--- ---- ---- ●--- ---- -●-- ---- ---- -1-- ---- ---- -1--
X_{18}^S	---- ●●●● ---- ---- ●●●● ---- ---- ●●●● ---- ●●●● ---- ---- ●●●● ---- ---- ●●●● ---- ●●●● ---- ---- ●●●● ---- ---- ●●●● ---- ●●●● ---- ---- ●11● ---- ---- ●11●
RK'_{18}	-- 11 -- -- 11 -- -- 11 -- 11 -- -- 11 -- -- 11 -- 11 -- -- 11 -- -- 11 -- 11 -- -- -- -- -- --
$X_{18}^{S,K}$	---- ●●●● ---- ---- ●●●● ---- ---- ●●●● ---- ●●●● ---- ---- ●●●● ---- ---- ●●●● ---- ●●●● ---- ---- ●●●● ---- ---- ●●●● ---- ●●●● ---- ---- ●11● ---- ---- ●11●
X_{19}	--●- ●●-- --●- ●●-- --●- ●●-- --●- ●1-- ---● -●●- ---● -●●- ---● -●●- ---● -11- ●--- --●● ●--- --●● ●--- --●● ●--- --1● -●-- ●--● -●-- ●--● -●-- ●--● -●-- ●--●

4.1.1 Searching Differential-Linear Approximations

The specialized MILP model is applied to assist in searching the differential-linear distinguishers for message processing phase of `GIFT-COFB`. To make the analysis simple and get a better result, the input difference and output linear mask of E_m are restricted to be single-bit. The correlation of differential-linear approximation of E_m is theoretically estimated according to the propagation of correlation of DL approximation for GS as shown in Proposition 1. After constructing the partial DLCT of E_m, the specialized MILP model is obtained to search for differential-linear approximations. In the test phase, no R-round differential-linear approximation with a correlation greater than 2^{-32} is found in our model when $R \geq 14$. Targeting for 13-round cipher E, with setting $r_d = 3$, $r_m = 8$, and $r_l = 2$, we can find better differential-linear distinguishers. In the key-recovery phase, with some subkey bits guessed, the distinguishers can be extended backward by three rounds and appended forward by two rounds. Note that the differential-linear distinguisher with the greatest correlation may not lead to the best key-recovery attack. Therefore, to obtain an attack as good as possible, we store thousands of distinguishers with high correlation and find the optimal one that has the lowest complexities in the key-recover attack. In Gurobi, the function $PoolSolution$ is used to find the l-best solution.

Therefore, we collect the top $l = 1024$ differential-linear distinguishers with a correlation greater than 2^{-32}. When we extend three rounds at the top and append two rounds at the bottom of these distinguishers, the minimum number of guessed subkey bits is 39. Only one differential-linear distinguisher achieves the minimum of guessed subkey bits.

As a result, we exploit the 13-round differential-linear approximation, with a correlation of $2^{-29.76}$, which has the minimum number 39 of guessed subkey bits, and the indices of active bits in its input difference are $\text{Index}_{\Delta_{in}} = \{84, 92, 94\}$, the indices of active bits in its output linear mask are $\text{Index}_{\Gamma_{out}} = \{10, 59, 105\}$. The differential-linear distinguisher is constructed by an 8-round differential-linear approximation of E_m with correlation $2^{-12.76}$, a 3-round differential trail of E_d with probability 2^{-11} and a 2-round linear trail of E_l with correlation 2^{-3}. The two trails are shown in the tables in the full paper. By the sampling experiment with 2^{33} random plaintext pairs for each of the 2^3 random keys, we have checked the correlation of the 8-round E_m, which is about $2^{-11.78}$ with input difference at the 95-th bit and output linear mask at the 39-th bit. Therefore, the correlation of 13-round differential-linear is estimated as $2^{-11} \times 2^{-11.78} \times 2^{-3\times 2} = 2^{-28.78}$ which is used in the following.

4.1.2 Key-recovery Attack

With the 13-round differential-linear approximation, an 18-round key-recovery attack on `GIFT-COFB` is given by appending three rounds at the top and two rounds at the bottom of this distinguisher. As illustrated in Table 4, the key-recovery attack is described by the following procedure where the 39 guessed key bits during the attack are listed in Table 5. In Table 4, the bit ordering is first from right to left, then from down to top. The symbol **-** indicates the inactive bits of the state. In the differential trail propagation, '$*$' denotes an uncertain bit of difference, and '1' denotes an active bit of difference. In the linear trail propagation, '$\bullet$' indicates a bit whose value needs to be computed, and '1' indicates a bit linearly involved.

1. Select $2N$ plaintexts, consisting of $\frac{2N}{2^{32}}$ structures, each is chosen by selecting:

   (a) Any intermediate X_1^P, and the remaining $2^{32} - 1$ intermediate values which differ from X_1^P by all the other $2^{32} - 1$ possibilities of the 32 bits which enter the 8 active S-boxes in round 1, i.e., $\{64 : 79, 96 : 111\}$.

   (b) The corresponding plaintexts are obtained by applying the transformation $SubCells^{-1} \circ PermBits^{-1}(\cdot)$ to the above 2^{32} intermediate values.

2. Request the ciphertexts of these plaintext structures (encrypted under the unknown key K).

3. For all the possible values of the 22-bit subkey $RK_1[\text{Index}_{RK_1}]||RK_2[\text{Index}_{RK_2}]$ (16 bits entering the 8 S-boxes in round 1 and 6 bits entering the 3 S-boxes in round 2), where $\text{Index}_{RK_1} = \{32:39, 48:55\}$ and $\text{Index}_{RK_2} = \{44, 45, 56, 57, 60, 61\}$,

    (a) Partially encrypt for each plaintext the 8 active S-boxes in round 1 and 3 active S-boxes in round 2, and find the pairs which satisfy the difference $\Delta X_4 = \Delta_{in}$ before round 3.

    (b) Given those N pairs, for all the possible values of the 17-bit subkey RK'_{17} $[\text{Index}_{RK'_{17}}]||RK'_{18}[\text{Index}_{RK'_{18}}]$ (extra 2 bits in round 17 and extra 15 bits in round 18), initialize a counter CT for the targeted parity $\Delta t = \oplus_{i \in \text{Index}_{X_{17}}} X_{17}[i]$, for each ciphertext pair, perform partial decryption on the 10 active S-boxes in round 18 and 2 S-boxes in round 17 and compute the value of Δt, if $\Delta t = 0$, then increase the counter CT by one, where where $\text{Index}_{RK'_{17}} = \{52, 53\}$, $\text{Index}_{RK'_{18}} = \{12, 13, 16, 17, 22, 23, 28, 29, 32, 39, 44, 45, 48, 60, 61\}$ and $\text{Index}_{X_{17}} = \{10, 59, 105\}$. If $|CT/N - 0.5| > \theta$, accept the current value of 39-bit subkey as a candidate.

4. The rest of the key bits are then recovered by exhaustively searching.

Complexity analysis. We set the advantage of attack as $a = 26$ to make a balance between the exhaustive search. When the data complexity is $D = 2N = 2^{64}$, the success probability is 85.23%. Therefore, the time complexity of procedure is $T = 2^{22} \times (2^{17} \times 2N) \times \frac{12}{18 \times 32} + 2^{128-a} = 2^{102.06}$.

Table 5: The 39 guessed key bits for the message processing phase of 18-round GIFT-COFB

1	$k_{91}k_{27}$ $k_{90}k_{26}$ $k_{89}k_{25}$ $k_{88}k_{24}$ $k_{83}k_{19}$ $k_{82}k_{18}$ $k_{81}k_{17}$ $k_{80}k_{16}$
2	$k_{126}k_{62}$ $k_{124}k_{60}$ $k_{118}k_{54}$
17	$k_{70}k_{22}$ k_{19}
18	$k_{103}k_{55}$ $k_{118}k_{62}$ $k_{126}k_{46}$ $k_{101}k_{53}$ $k_{116}k_{60}$ $k_{124}k_{44}$ $k_{99}k_{51}$ $k_{114}k_{58}$ $k_{122}k_{42}$ $k_{97}k_{49}$

4.2 Attack on Initialization Phase

The attack on the initialization phase of GIFT-COFB is similar to the one on the encryption procedure in the message processing phase. For searching differential-linear distinguishers, compared with the one for the message processing phase, there is no restriction on active bits on the input difference of distinguisher for the initialization phase. In the following, we first give the new differential-linear distinguisher, and then the key-recovery attack is presented based on it.

Differential-linear distinguisher. With the automatic tool, we found a 13-round differential-linear approximation with correlation of $2^{-27.78}$, whose indices of active bits in its input difference are $\text{Index}_{\Delta_{in}} = \{86, 87, 94, 95\}$ and output linear mask are $\text{Index}_{\Gamma_{out}} = \{10, 59, 105\}$. Compared with the differential-linear distinguisher for the message processing phase, the 13-round distinguisher has a different 3-round differential trail of E_d with probability 2^{-10}, and the same 8-round differential-linear approximation of E_m and the 2-round linear trail of E_l. The new 3-round differential of E_d is shown in the full paper.

Key-recovery attack. Based on the above 13-round differential-linear approximation, an 18-round key-recovery attack is given on the initialization phase of GIFT-COFB by extending three rounds at the top and appending two rounds at the bottom of this distinguisher. The key-recovery attack is presented in the full paper, where 41 key bits are guessed. For the details of the procedure of attack, please refer to the full paper.

Complexity analysis. The advantage of attack is set as $a = 35$ to make a balance between the exhaustive search. When the data complexity is $D = 2N = 2^{62.41}$, the success probability is 85.23%. Therefore, the time complexity of procedure is $T = 2^{24} \times (2^{17} \times 2N) \times \frac{12}{18\times 32} + 2^{128-a} = 2^{97.88}$.

5 Differential-Linear Attacks on HyENA

In this section, we first give the differential-linear distinguisher which is found by the specialized MILP for the message processing phase of HyENA. Then, based on the distinguisher, the key-recovery attacks are given on the message processing phase of HyENA.

For the attacks on the message processing phase, the target is the encryption procedure which is illustrated in the full paper. From the plaintext-ciphertext pair of HyENA $(M_0||M_1, C_0||C_1)$, we can get the input-output pair of the cipher GIFT-128, and the input is $(Y_a[64-127]||M_0[0-63]) \oplus (M_0[64-127]||2^{a+2}\Delta)$, the corresponding output is $M_1 \oplus C_1$, where $Y_a = M_0 \oplus C_0$. Under the nonce misusing scenario, another input-output pair can be chosen for GIFT-128. Due to the fact that the value of Δ is unknown, we can not determine the least significant 64 bits of input for GIFT-128. So, the data structure used in our key-recovery attacks can not involve the least significant 64 bits of the input for GIFT-128.

Similar to the case of GIFT-COFB, we can derive the following proposition for HyENA (actually the complement of Proposition 2, this result can be obtained in the same way).

Proposition 3. *Let ΔP be the input difference of plaintext, and ΔX_4 be the input difference of the 4-th round, i.e., the output difference of the 3-rd round. If Index$(\Delta X_4) \subseteq S'$, then Index$(\Delta P) \subseteq \{64, 65, \cdots, 127\}$ after three rounds backward; vice versa, where the Index$(\cdot)$ function returns the indices on which the value is non-zero, and $S' = \{4j_0, 4j_0 + 2 | j_0 \in \{8, 9, \cdots, 15, 24, 25, \cdots, 31\}\} \cup \{4j_1 + 1, 4j_1 + 3 | j_1 \in \{0, 1, \cdots, 7, 16, 17, \cdots, 23\}\}$.*

Therefore, we set the specialized MILP model to search DL approximations for HyENA such that there is no active bit in $\{0, 1, \cdots, 127\} \setminus S'$, where S' is defined in Proposition 3. From Proposition 3, the DL distinguishers returned by the specialized MILP model will always satisfy the restriction in the key-recovery attack.

5.1 Searching Differential-Linear Approximations

The specialized MILP model is applied to search differential-linear approximations for the message processing phase of HyENA. Similarly, the input difference and output linear mask of E_m are restricted to be single-bit. In the test phase, no R-round differential-linear approximation with a correlation greater than 2^{-32} is found in our MILP model when $R \geq 14$. As a result, we found a 13-round differential-linear approximation with correlation of $2^{-30.37}$, and the indices of active bits in its input difference are $\text{Index}_{\Delta_{in}} = \{96, 98, 104\}$, the indices of active bits in its output linear mask are $\text{Index}_{\Gamma_{out}} = \{49, 82, 99\}$. The differential-linear distinguisher is constructed by an 8-round differential-linear approximation of E_m with correlation $2^{-13.37}$, a 3-round differential trail of E_d with probability 2^{-11} and a

2-round linear trail of E_l with correlation 2^{-3}. The two trails are shown in the full paper. By the sampling experiment with 2^{33} random plaintext pairs for each of the 2^3 random keys, we have checked the correlation of the 8-round E_m, which is about $2^{-12.51}$ with input difference at the 19-th bit and output linear mask at the 3-rd bit. Therefore, the correlation of 13-round differential-linear is estimated as $2^{-11} \times 2^{-12.51} \times 2^{-3\times 2} = 2^{-29.51}$ which is used in the following.

Table 6: The attack on message processing phase of 18-round `HyENA`

ΔP	**** **** **** **** **** **** **** **** **** **** **** **** **** **** **** **** ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ----
ΔX_1^S	*--* **-- -**- --** *--* 1*-- -1*- --** *--* **-- -**- --** *--* **-- -**- --** ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ----
ΔX_1^P	---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- **** 11** **** **** ---- ---- ---- ---- **** **** **** **** ---- ---- ---- ----
RK_1	-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- 11 11 11 11 -- -- -- -- 11 11 11 11 -- -- -- --
ΔX_2	---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- **** 11** **** **** ---- ---- ---- ---- **** **** **** **** ---- ---- ---- ----
ΔX_2^S	---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- 1--- -1-- --*- ---* ---- ---- ---- ---- 1--* 1*-- -**- --** ---- ---- ---- ----
ΔX_2^P	---- 1*** ---- ---- ---- ---- ---- 11** ---- 1*** ----
RK_2	-- 11 -- -- -- -- -- 11 -- 11 --
ΔX_3	---- 1*** ---- ---- ---- ---- ---- 11** ---- 1*** ----
ΔX_3^S	---- ---1 ---- ---- ---- ---- ---- -1-- ---- ---1 ----
ΔX_3^P	---- ---- ---- ---- ---- ---1 ---- -1-1 ----
RK_3	-- --
Δ_{in}	---- ---- ---- ---- ---- ---1 ---- -1-1 ----
Γ_{out}	---- ---- ---- ---- ---- ---- ---- 1--- ---- ---- ---- -1-- ---- ---- ---- ---- ---- ---- ---- --1- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ----
X_{17}^S	---- ---- ---- ---- ---- ---- ---- ●1●1 ---- ---- ---- ●11● ---- ---- ---- ---- ---- ---- ---- ●●●● ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ----
RK'_{17}	-- -- -- -- -- -- -- -1 -- -- -- -- -- -- -- -- -- -- -- 11 -- -- -- -- -- -- -- -- -- -- -- --
$X_{17}^{S,K}$	---- ---- ---- ---- ---- ---- ---- ●1●1 ---- ---- ---- ●11● ---- ---- ---- ---- ---- ---- ---- ●●●● ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ----
X_{18}	---- ●--- ●--- ---- ●--- ---- ---- ---- ---- -1-- -1-- ---- -●-- ---- ---- ---- ---- --●- --1- ---- --●- ---- ---- ---- ---- ---1 ---● ---- ---● ---- ---- ----
X_{18}^S	---- ●●●● ●●●● ---- ●●●● ---- ---- ---- ---- ●11● ●11● ---- ●●●● ---- ---- ---- ---- ●●●● ●●●● ---- ●●●● ---- ---- ---- ---- ●●●● ●●●● ---- ●●●● ---- ---- ----
RK'_{18}	-- 11 11 -- 11 -- -- -- -- -- -- -- 11 -- -- -- -- 11 11 -- 11 -- -- -- -- 11 11 -- 11 -- -- --
$X_{18}^{S,K}$	---- ●●●● ●●●● ---- ●●●● ---- ---- ---- ---- ●11● ●11● ---- ●●●● ---- ---- ---- ---- ●●●● ●●●● ---- ●●●● ---- ---- ---- ---- ●●●● ●●●● ---- ●●●● ---- ---- ----
X_{19}	--●● -●-- --1● -●-- --●● -●-- --●● -●-- ●--● --●- ●--● --●- ●--● --●- ●--● --●- ●●-- ---● ●1-- ---● ●●-- ---● ●●-- ---● -●●- ●--- -11- ●--- -●●- ●--- -●●- ●---

5.2 Key-recovery Attack

With 13-round differential-linear approximation, we give an 18-round key-recovery attack on `HyENA` by extending three rounds at the top and appending two rounds at the bottom of the distinguisher. The key-recovery attack is illustrated in Table 6 where 39 key bits are guessed. For the detailed procedure of the attack, please refer to the full paper.

Complexity analysis. The advantage of attack is set as $a = 9$ to make a balance between the exhaustive search. When the data complexity is $D = 2N = 2^{63.97}$, the success probability is 85.21%. Therefore, the time complexity of the procedure is $T = 2^{22} \times (2^{17} \times 2N) \times \frac{12}{18\times 32} + 2^{128-a} = 2^{119}$.

Remark. The attack on the initialization phase of `HyENA` is similar to the one on the message processing phase. Compared with the one for the message processing phase, the input difference can be imposed at the most 96 significant bits. Therefore, the attack on the message processing phase can also be launched for the initialization phase of `HyENA`.

6 Differential-Linear Cryptanalysis of `GIFT-64/128`

To promote a comprehensive perception of the soundness of `GIFT-64/128`, we evaluated the security `GIFT-64/128` against differential-linear attacks in this section.

6.1 Attack on `GIFT-64`

First, a 12-round differential-linear approximation is found, and then we give an 18-round key-recovery attack on `GIFT-64` based on the DL approximation.

Searching differential-linear approximation. To simplify, the input difference and output linear mask of E_m are restricted to be single-bit. In the test phase, we could not find the R-round differential-linear approximation correlation greater than 2^{-32} when $R \geq 13$. For 12-round `GIFT-64`, with setting $r_d = 2$, $r_m = 7$ and $r_l = 3$, the better differential-linear distinguishers was found. With the automatic tool, we found a 13-round differential-linear approximation with correlation of $2^{-28.61}$, whose indices of active bits in its input difference are $\text{Index}_{\Delta_{in}} = \{34, 35, 38, 39\}$, the indices of active bits in its output linear mask are $\text{Index}_{\Gamma_{out}} = \{20, 30, 41, 54, 58, 60\}$. The differential-linear distinguisher consists of a 7-round differential-linear approximation of E_m with correlation $2^{-10.61}$, a 2-round differential trail of E_d with probability 2^{-6} and a 3-round linear trail of E_l with correlation 2^{-6} which are shown in the full paper. The theoretical estimation of correlation $2^{-28.61}$ is used in the following analysis of attack complexity.

Key-recovery attack. Based on the above 12-round differential-linear distinguisher, an 18-round key-recovery attack is given by appending three rounds at the top and three rounds at the bottom of this distinguisher. The key-recovery attack on 18-round `GIFT-64` is given in the full paper, where 66 key bits are guessed.

Complexity analysis. The advantage of attack is set as $a = 6$ to make a balance between the exhaustive search. When the data complexity is $D = 2N = 2^{61.57}$, the success probability is 85.07%. The time complexity of the procedure is $T = 2^{66} \times 2N \times \frac{31}{18\times 16} + 2^{128-a} = 2^{124.61}$.

6.2 Attack on `GIFT-128`

In this section, we present a key-recovery attack on 19 rounds of `GIFT-128` which is based on a 17-round differential-linear approximation. The differences between on `GIFT-128` and `GIFT-COFB` are no data limitation of 2^{64} but less than the space of entire block size 2^{128} and no restriction of the input difference on the least significant 64 bits for `GIFT-128`.

Searching differential-linear approximation. The input difference and output linear mask of E_m are restricted to be single-bit. In the test phase, we did not find $R \geq 18$-round differential-linear approximation with a correlation greater than 2^{-64}. With setting $r_d = 4$, $r_m = 8$ and $r_l = 5$, a differential-linear distinguisher is constructed for 17-round `GIFT-128`. The 17-round differential-linear approximation with correlation of $2^{-58.78}$, whose indices of active bits in its input difference are $\text{Index}_{\Delta_{in}} = \{82, 83, 93, 94, 121, 122, 123\}$, the indices of active bits in its output linear mask are
$\text{Index}_{\Gamma_{out}} = \{50, 54, 91, 95, 112, 116\}$. The 17-round differential-linear distinguisher consists of an 8-round differential-linear approximation of E_m with correlation $2^{-18.78}$, a 4-round differential trail of E_d with probability 2^{-16} and a 5-round linear trail of E_l with correlation 2^{-12}, which are shown in the full paper. The theoretical estimation of correlation $2^{-58.78}$ is used in the following analysis of attack complexity.

Key-recovery attack. Based on the above 17-round differential-linear distinguisher, a 19-round key-recovery attack is given by extending one round at the top and one round at the bottom of this distinguisher. The key-recovery attack on 19-round `GIFT-128` is given in the full paper, where 6 key bits are guessed.

Complexity analysis. The advantage of attack is set as $a = 9$ to make a balance between the exhaustive search. When the data complexity is $D = 2N = 2^{122.51}$, and the success probability is 85.21%. The time complexity of the procedure is $T = 2^6 \times 2N \times \frac{4}{19\times 32} + 2^{128-a} = 2^{121.53}$.

7 Conclusion

In this paper, we evaluated the security of `GIFT-64/128`, `GIFT-COFB` and `HyENA` against differential-linear cryptanalysis. The automatic tool was developed for searching differential-linear approximations for the ciphers based on S-boxes. With the application of our automatic tool, we found the 13-round differential-linear distinguishers for `GIFT-COFB` and `HyENA`, and the 18-round key-recovery attacks were given on both ciphers, which cover two rounds more than the previous best ones. As regards `GIFT-64` and `GIFT-128`, the 12-round and 17-round differential-linear distinguishers were found, leading to the 18-round and 19-round key-recovery attacks respectively. The attacks on `GIFT-64` and `GIFT-128` could not reach the same rounds with the best attacks obtained by the differential cryptanalysis in [CZD19] and [ZDC+21] respectively, same to the linear case. We stress again that our attacks do not threaten the security of these ciphers.

In future work, we will continue to improve the automatic tool for differential-linear cryptanalysis. Although the (differential-linear) distinguishers with more rounds are found, fewer rounds are appended at the top and bottom at the distinguishers to launch the key-recovery attacks. Therefore, more advanced techniques may improve further the key-recovery attacks in differential-linear cryptanalysis, such as the fast Fourier transform (FFT) and filtering technique with guessing partial S-boxes. Another one is how to integrate the key-recovery part into the MILP model. This strategy could be used in our attacks, and we will further investigate how it could improve the results. Furthermore, we are going to analyze other ciphers and evaluate their security with the automatic tool.

Acknowledgements. We are very grateful to Willi Meier for his detailed comments and helpful suggestions for our manuscript. We would like to thank the anonymous reviewers for their helpful comments and useful discussions of the manuscript.

References

[BCI$^+$21] Subhadeep Banik, Avik Chakraborti, Tetsu Iwata, Kazuhiko Minematsu, Mridul Nandi, Thomas Peyrin, Yu Sasaki, Siang Meng Sim, and Yosuke Todo. GIFT-COFB. *NIST Lightweight Cryptography Project*, 2021. https://csrc.nist.gov/Projects/lightweight-cryptography/finalists. URL: https://csrc.nist.gov/Projects/lightweight-cryptography/finalists.

[BDKW19] Achiya Bar-On, Orr Dunkelman, Nathan Keller, and Ariel Weizman. DLCT: A new tool for differential-linear cryptanalysis. In Yuval Ishai and Vincent Rijmen, editors, *Advances in Cryptology - EUROCRYPT 2019 - 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Darmstadt, Germany, May 19-23, 2019, Proceedings, Part I*, volume 11476 of *Lecture Notes in Computer Science*, pages 313–342. Springer, 2019. doi:10.1007/978-3-030-17653-2_11.

[BGG$^+$23] Emanuele Bellini, David Gérault, Juan Grados, Rusydi H. Makarim, and Thomas Peyrin. Fully automated differential-linear attacks against ARX ciphers. In Mike Rosulek, editor, *Topics in Cryptology - CT-RSA 2023 - Cryptographers' Track at the RSA Conference 2023, San Francisco, CA, USA, April 24-27, 2023, Proceedings*, volume 13871 of *Lecture Notes in Computer Science*, pages 252–276. Springer, 2023. doi:10.1007/978-3-031-30872-7_10.

[BLN17] Céline Blondeau, Gregor Leander, and Kaisa Nyberg. Differential-linear cryptanalysis revisited. *J. Cryptol.*, 30(3):859–888, 2017. doi:10.1007/s00145-016-9237-5.

[BLT20] Christof Beierle, Gregor Leander, and Yosuke Todo. Improved differential-linear attacks with applications to ARX ciphers. In Daniele Micciancio and Thomas Ristenpart, editors, *Advances in Cryptology - CRYPTO 2020 - 40th Annual International Cryptology Conference, CRYPTO 2020, Santa Barbara, CA, USA, August 17-21, 2020, Proceedings, Part III*, volume 12172 of *Lecture Notes in Computer Science*, pages 329–358. Springer, 2020. doi:10.1007/978-3-030-56877-1_12.

[BPP$^+$17] Subhadeep Banik, Sumit Kumar Pandey, Thomas Peyrin, Yu Sasaki, Siang Meng Sim, and Yosuke Todo. GIFT: A small present - towards reaching the limit of lightweight encryption. In Wieland Fischer and Naofumi Homma, editors, *Cryptographic Hardware and Embedded Systems - CHES 2017 - 19th International Conference, Taipei, Taiwan, September 25-28, 2017, Proceedings*, volume 10529 of *Lecture Notes in Computer Science*, pages 321–345. Springer, 2017. doi:10.1007/978-3-319-66787-4_16.

[BS90] Eli Biham and Adi Shamir. Differential cryptanalysis of des-like cryptosystems. In Alfred Menezes and Scott A. Vanstone, editors, *Advances in Cryptology - CRYPTO '90, 10th Annual International Cryptology Conference, Santa Barbara, California, USA, August 11-15, 1990, Proceedings*, volume 537 of *Lecture Notes in Computer Science*, pages 2–21. Springer, 1990. doi:10.1007/3-540-38424-3_1.

[CDJN19] Avik Chakraborti, Nilanjan Datta, Ashwin Jha, and Mridul Nandi. HyENA. *NIST Lightweight Cryptography Project*, 2019. https://csrc.nist.gov/Projects/lightweight-cryptography/round-2-candidates. URL:

https://csrc.nist.gov/projects/lightweight-cryptography/round-2-candidates.

[CN21] Murilo Coutinho and Tertuliano C. Souza Neto. Improved linear approximations to ARX ciphers and attacks against chacha. In Anne Canteaut and François-Xavier Standaert, editors, *Advances in Cryptology - EUROCRYPT 2021 - 40th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, October 17-21, 2021, Proceedings, Part I*, volume 12696 of *Lecture Notes in Computer Science*, pages 711–740. Springer, 2021. doi:10.1007/978-3-030-77870-5_25.

[CZD19] Huaifeng Chen, Rui Zong, and Xiaoyang Dong. Improved differential attacks on GIFT-64. In Jianying Zhou, Xiapu Luo, Qingni Shen, and Zhen Xu, editors, *Information and Communications Security - 21st International Conference, ICICS 2019, Beijing, China, December 15-17, 2019, Revised Selected Papers*, volume 11999 of *Lecture Notes in Computer Science*, pages 447–462. Springer, 2019. doi:10.1007/978-3-030-41579-2_26.

[HPTY23] Kai Hu, Thomas Peyrin, Quan Quan Tan, and Trevor Yap. Revisiting higher-order differential-linear attacks from an algebraic perspective. In Jian Guo and Ron Steinfeld, editors, *Advances in Cryptology - ASIACRYPT 2023 - 29th International Conference on the Theory and Application of Cryptology and Information Security, Guangzhou, China, December 4-8, 2023, Proceedings, Part III*, volume 14440 of *Lecture Notes in Computer Science*, pages 405–435. Springer, 2023. doi:10.1007/978-981-99-8727-6_14.

[JZZD20a] Fulei Ji, Wentao Zhang, Chunning Zhou, and Tianyou Ding. Improved (related-key) differential cryptanalysis on GIFT. *IACR Cryptol. ePrint Arch.*, page 1242, 2020. URL: https://eprint.iacr.org/2020/1242.

[JZZD20b] Fulei Ji, Wentao Zhang, Chunning Zhou, and Tianyou Ding. Improved (related-key) differential cryptanalysis on GIFT. In Orr Dunkelman, Michael J. Jacobson Jr., and Colin O'Flynn, editors, *Selected Areas in Cryptography - SAC 2020 - 27th International Conference, Halifax, NS, Canada (Virtual Event), October 21-23, 2020, Revised Selected Papers*, volume 12804 of *Lecture Notes in Computer Science*, pages 198–228. Springer, 2020. doi:10.1007/978-3-030-81652-0_8.

[Leu16] Gaëtan Leurent. Improved differential-linear cryptanalysis of 7-round chaskey with partitioning. In Marc Fischlin and Jean-Sébastien Coron, editors, *Advances in Cryptology - EUROCRYPT 2016 - 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, May 8-12, 2016, Proceedings, Part I*, volume 9665 of *Lecture Notes in Computer Science*, pages 344–371. Springer, 2016. doi:10.1007/978-3-662-49890-3_14.

[LH94] Susan K. Langford and Martin E. Hellman. Differential-linear cryptanalysis. In Yvo Desmedt, editor, *Advances in Cryptology - CRYPTO '94, 14th Annual International Cryptology Conference, Santa Barbara, California, USA, August 21-25, 1994, Proceedings*, volume 839 of *Lecture Notes in Computer Science*, pages 17–25. Springer, 1994. doi:10.1007/3-540-48658-5_3.

[LJC23] Guangqiu Lv, Chenhui Jin, and Ting Cui. A miqcp-based automatic search algorithm for differential-linear trails of ARX ciphers(long paper). *IACR Cryptol. ePrint Arch.*, page 259, 2023. URL: https://eprint.iacr.org/2023/259.

[LLL21] Meicheng Liu, Xiaojuan Lu, and Dongdai Lin. Differential-linear cryptanalysis from an algebraic perspective. In Tal Malkin and Chris Peikert, editors, *Advances in Cryptology - CRYPTO 2021 - 41st Annual International Cryptology Conference, CRYPTO 2021, Virtual Event, August 16-20, 2021, Proceedings, Part III*, volume 12827 of *Lecture Notes in Computer Science*, pages 247–277. Springer, 2021. doi:10.1007/978-3-030-84252-9_9.

[LSL21] Yunwen Liu, Siwei Sun, and Chao Li. Rotational cryptanalysis from a differential-linear perspective - practical distinguishers for round-reduced friet, xoodoo, and alzette. In Anne Canteaut and François-Xavier Standaert, editors, *Advances in Cryptology - EUROCRYPT 2021 - 40th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, October 17-21, 2021, Proceedings, Part I*, volume 12696 of *Lecture Notes in Computer Science*, pages 741–770. Springer, 2021. doi:10.1007/978-3-030-77870-5_26.

[Mat93] Mitsuru Matsui. Linear cryptanalysis method for DES cipher. In Tor Helleseth, editor, *Advances in Cryptology - EUROCRYPT '93, Workshop on the Theory and Application of of Cryptographic Techniques, Lofthus, Norway, May 23-27, 1993, Proceedings*, volume 765 of *Lecture Notes in Computer Science*, pages 386–397. Springer, 1993. doi:10.1007/3-540-48285-7_33.

[MWGP11] Nicky Mouha, Qingju Wang, Dawu Gu, and Bart Preneel. Differential and linear cryptanalysis using mixed-integer linear programming. In Chuankun Wu, Moti Yung, and Dongdai Lin, editors, *Information Security and Cryptology - 7th International Conference, Inscrypt 2011, Beijing, China, November 30 - December 3, 2011. Revised Selected Papers*, volume 7537 of *Lecture Notes in Computer Science*, pages 57–76. Springer, 2011. doi:10.1007/978-3-642-34704-7_5.

[NSLL22] Zhongfeng Niu, Siwei Sun, Yunwen Liu, and Chao Li. Rotational differential-linear distinguishers of ARX ciphers with arbitrary output linear masks. In Yevgeniy Dodis and Thomas Shrimpton, editors, *Advances in Cryptology - CRYPTO 2022 - 42nd Annual International Cryptology Conference, CRYPTO 2022, Santa Barbara, CA, USA, August 15-18, 2022, Proceedings, Part I*, volume 13507 of *Lecture Notes in Computer Science*, pages 3–32. Springer, 2022. doi:10.1007/978-3-031-15802-5_1.

[Sel08] Ali Aydin Selçuk. On probability of success in linear and differential cryptanalysis. *J. Cryptol.*, 21(1):131–147, 2008. doi:10.1007/s00145-007-9013-7.

[SHW$^+$14a] Siwei Sun, Lei Hu, Meiqin Wang, Peng Wang, Kexin Qiao, Xiaoshuang Ma, Danping Shi, Ling Song, and Kai 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. Cryptology ePrint Archive, Report 2014/747, 2014. https://ia.cr/2014/747.

[SHW$^+$14b] Siwei Sun, Lei Hu, Peng Wang, Kexin Qiao, Xiaoshuang Ma, and Ling Song. Automatic security evaluation and (related-key) differential characteristic search: Application to simon, present, lblock, DES(L) and other bit-oriented block ciphers. In Palash Sarkar and Tetsu Iwata, editors, *Advances in Cryptology - ASIACRYPT 2014 - 20th International Conference on the Theory and Application of Cryptology and Information Security, Kaoshiung, Taiwan, R.O.C., December 7-11, 2014. Proceedings, Part I*, volume 8873 of *Lecture Notes in Computer Science*, pages 158–178. Springer, 2014. doi:10.1007/978-3-662-45611-8_9.

[SWW21a] Ling Sun, Wei Wang, and Meiqin Wang. Improved attacks on GIFT-64. In Riham AlTawy and Andreas Hülsing, editors, *Selected Areas in Cryptography - 28th International Conference, SAC 2021, Virtual Event, September 29 - October 1, 2021, Revised Selected Papers*, volume 13203 of *Lecture Notes in Computer Science*, pages 246–265. Springer, 2021. doi:10.1007/978-3-030-99277-4_12.

[SWW21b] Ling Sun, Wei Wang, and Meiqin Wang. Linear cryptanalyses of three aeads with GIFT-128 as underlying primitives. *IACR Trans. Symmetric Cryptol.*, 2021(2):199–221, 2021. doi:10.46586/tosc.v2021.i2.199-221.

[SWW22] Ling Sun, Wei Wang, and Meiqin Wang. Addendum to linear cryptanalyses of three aeads with GIFT-128 as underlying primitives. *IACR Trans. Symmetric Cryptol.*, 2022(1):212–219, 2022. URL: https://doi.org/10.46586/tosc.v2022.i1.212-219, doi:10.46586/TOSC.V2022.I1.212-219.

[ZDC$^+$21] Rui Zong, Xiaoyang Dong, Huaifeng Chen, Yiyuan Luo, Si Wang, and Zheng Li. Towards key-recovery-attack friendly distinguishers: Application to GIFT-128. *IACR Trans. Symmetric Cryptol.*, 2021(1):156–184, 2021. doi:10.46586/tosc.v2021.i1.156-184.

[ZDY19] Baoyu Zhu, Xiaoyang Dong, and Hongbo Yu. Milp-based differential attack on round-reduced GIFT. In Mitsuru Matsui, editor, *Topics in Cryptology - CT-RSA 2019 - The Cryptographers' Track at the RSA Conference 2019, San Francisco, CA, USA, March 4-8, 2019, Proceedings*, volume 11405 of *Lecture Notes in Computer Science*, pages 372–390. Springer, 2019. doi:10.1007/978-3-030-12612-4_19.