MILP Based Differential Attack on Round Reduced WARP

Manoj Kumar and Tarun Yadav

Scientific Analysis Group, DRDO, Metcalfe House Complex, Delhi-110054, INDIA
{manojkumar,tarunyadav}@sag.drdo.in

Abstract. WARP is proposed by S. Banik et al. in SAC 2020. It is a 128-bit lightweight block cipher with 128-bit key. WARP is based on the 32-nibble type-2 Generalised Feistel Network (GFN) structure. It uses a permutation over nibbles which is designed to optimize the security and efficiency. The designers have provided a lower bound for the number of differentially active S-boxes but the detailed differential characteristics are not provided. In this paper, we discuss the MILP based search technique and present the differential characteristics for the 18-round and 19-round WARP with probability of 2^{-122} and 2^{-132} respectively. We also present a key recovery attack on the 21-round WARP with data complexity of 2^{113} chosen plaintexts. To the best of our knowledge, these detailed differential characteristics are presented for the first time and this is the first key recovery attack on the 21-round WARP.

Keywords: Lightweight Cryptography, Block Cipher, Differential Cryptanalysis, MILP

1 Introduction

Lightweight cryptography is used for encryption as well as authentication on small computing devices *e.g.* RFID tags, sensor networks and smart cards [5]. Lightweight block cipher PRESENT is the first notable design which was published in 2007 [3]. Plenty of lightweight block ciphers are designed in the past two decades. Initially, the 64-bit block with a key size of 80/128 bits was used for designing the lightweight version of block ciphers. Nowadays, the 64/128 bits block with a 128-bit key is preferred to design the lightweight block cipher. The 128-bit lightweight block ciphers can serve as good candidates to replace the AES [4] where not only the security but also the computational complexity is a major constraint. NIST has initiated a competition in 2018 to standardise the lightweight cryptographic algorithms seeing the increasing importance of lightweight cryptography. Therefore, the security analysis of the lightweight block ciphers is required to assess the strength against the basic cryptanalytic attacks.

The differential attack is a basic cryptanalysis technique proposed by E. Biham and A. Shamir [2] in 1990. This exploits the non-uniform relations between

the input and output differences. The probability of the best differential characteristic is used to provide a bound on the security of block cipher against the differential attack. High probability differential characteristics are essential for a successful key recovery attack using the differential cryptanalysis technique. The techniques based on the automated search are used to construct these differential characteristics. M. Matsui [7] proposed a branch-and-bound based technique to search the high probability differential characteristics in 1993. This technique has some limitations to search the differential characteristics for large block sizes. In 2012, N. Mohua et al. [8] proposed a new technique using the Mixed Integer Linear Programming (MILP) to search the differential characteristics more efficiently.

MILP deals with the optimization problems in which the objective function and the constraints are linear. There are various commercial linear programming problem (LPP) solvers *e.g.* Gurobi [13] and CPLEX [14]. These solvers provide the solution for an LPP problem very efficiently. Mouha et al. proposed a framework to convert the differential characteristic search problem into an MILP problem and used these MILP solvers to provide the characteristics with least number of active S-boxes. At Asiacrypt 2014, Sun et al. [10] applied the MILP based attack on the bit oriented block ciphers using the H-Representation of convex hull for all differential patterns of the S-box to find the differential characteristics. The differential characteristic search problem is divided into two modules. In first module, a lower bound on the number of differentially active S-boxes is computed. While, the differential characteristics with high probability are constructed in the second module. The similar kind of differential attack was published by B. Zhu et al. on the lightweight block cipher GIFT [11].

The designers of WARP [1] have also provided a security bound against the differential attack. They used the MILP-aided search to compute a lower bound for the number of differentially active S-boxes. But, they have not provided the differential characteristics with these bounds. According to the designers analysis, there are total 61 active S-boxes in any 18-round differential characteristics of WARP which can be used for the key recovery. Similar bound for the 19-round differential characteristic is given as 66 which requires 2^{132} chosen plain text pairs and it is infeasible for a 128-bit block cipher. In this paper, we construct the differential characteristics for the 18-round WARP using the MILP-aided search. Firstly, we compute a lower bound on the number of differentially active S-boxes which is equal to the designers bound. Secondly, we construct the actual differential characteristics for the 18 and 19 rounds of WARP with probability of 2^{-122} and 2^{-132} respectively. We also present a key recovery attack on the 21-round WARP which is the best differential attack against WARP till date.

We organise the remaining paper in the following manner. In Section 2, we provide a brief introduction to the lightweight block cipher WARP. In Section 3, we compute a lower bound on the number of differentially active S-boxes and construct the 18-round and 19-round differential characteristics using the MILP-aided search. We provide a key recovery attack on the 21-round WARP in Section 4. The paper is concluded in Section 5.

2 Description of WARP

The base structure of the lightweight block cipher WARP is a type-2 Generalised Feistel Network (GFN) structure. There are many 64-bit block ciphers, with 16 branches, designed using the type-2 GFN structure. But, slow diffusion in 64-bit block with 16 branches is a security challenge. The GFN is revisited by S. Banik et al. [1] and 128-bit block size with 32-branches is considered more suitable to design a 128-bit lightweight block cipher.

2.1 Encryption Algorithm:

WARP encrypts the 128-bit plaintext block using a 128-bit key and generates a 128-bit ciphertext block. There are total 41 rounds. The designers have explained the round function of WARP in various equivalent forms. We have used the LBlock like equivalent form of WARP to describe its encryption process. The encryption algorithm encrypts 128-bit the input X using a 128-bit key K (Algorithm 1). The key expansion algorithm is not required for WARP. The Key K is divided into two 64-bit keys K_0 & K_1 and it is expressed as $K = (K_0, K_1)$. In the odd rounds, the left part K_0 is used and every even round uses the right part K_1. We express the 128-bit input X using 32 nibbles starting from right to left. The initial permutation (IP) is applied on X to get two 64-bit words X_{2i} and X_{2i+1} (for $0 \leq i \leq 15$). In each round, the constants (Table 1) are XORed with the first two nibbles of X^r_{2i+1}. Thereafter, the S-box layer (Table 2) is applied on X^r_{2i} by dividing it in 4-bit nibbles. Then, the output of S-box layer is XORed with the round key. The nibble permutation N_P (Table 3) is applied thereafter to get a 64-bit output U. The cyclic rotation by 24 bits is applied on X^r_{2i+1} to get a 64-bit output V. To get X^{r+1}_{2i}, U and V are XORed while X^r_{2i} becomes X^{r+1}_{2i+1} due to the Feistel structure. This process is applied 40 times iteratively and the last round is performed without the rotation and permutation operations.

Algorithm 1: Encryption Algorithm

Input: $X = (x_{31}, x_{30}, \cdots, x_0)$ and $K = (K_0, K_1)$
Output: X^{41}
IP: $X^1_{2i} = (x_0, x_2, \cdots, x_{30})$, $X^1_{2i+1} = (x_1, x_3, \cdots, x_{31})$, where $0 \leq i \leq 15$
for *r=1 to 40* **do**
  $X^r_1 = X^r_1 \oplus RC^r_0, X^r_3 = X^r_3 \oplus RC^r_1$
  $Y = S(X^r_{2i})$
  $U = N_P(Y \oplus K_{(r-1) mod 2})$
  $V = X^r_{2i+1} <<< 24$
  $X^{r+1}_{2i} = U \oplus V$
  $X^{r+1}_{2i+1} = X^r_{2i}$
end
$X^{41}_1 = X^{40}_1 \oplus RC^{40}_0, X^{41}_3 = X^{40}_3 \oplus RC^{40}_1$
$X^{41}_{2i} = X^{40}_{2i}$
$X^{41}_{2i+1} = S(X^{40}_{2i}) \oplus K_0 \oplus X^{40}_{2i+1}$

Round Constant: In each round, the 4-bit constants given in Table 1 are used.

r	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
RC_0^r	0	0	1	3	7	f	f	f	e	d	a	5	a	5	b	6	c	9	3	6	d
RC_1^r	4	c	c	c	c	c	8	4	8	4	8	4	c	8	0	4	c	8	4	c	c
r	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	
RC_0^r	b	7	e	d	b	6	d	a	4	9	2	4	9	3	7	e	c	8	1	2	
RC_1^r	8	4	c	8	4	8	0	4	8	0	4	c	c	8	0	0	4	8	4	c	

Table 1: Round Constants

S-box: The 4-bit S-box (Table 2) is applied in the S-box layer of WARP.

x	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
S(x)	c	a	d	3	e	b	f	7	8	9	1	5	0	2	4	6

Table 2: S-Box

Nibble Permutation: The output from the S-box layer is divided into 16 nibbles. The nibble permutation N_P is applied on these 16 nibbles (Table 3).

i	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$N_P(i)$	3	7	6	4	1	0	2	5	11	15	14	12	9	8	10	13

Table 3: Permutation

3 Differential Characteristics for 18 and 19 Rounds

3.1 Differential Cryptanalysis

The differential attack is a powerful cryptanalysis tool proposed by Biham and Shamir against DES [2] in 1990. In this attack, the propagation of input differences is studied to find the high probable output differences. These non-uniform relations are used as a distinguisher and the round subkeys are recovered using these distinguishers. Therefore, we need a differential characteristic suggesting the particular input and output occurrences with very high probability p for the target cipher. The data complexity of differential attack is inversely proportional to the probability p of a differential characteristic. Which means that we need p^{-1} chosen plaintext pairs to distinguish the r rounds of an n-bit block cipher.

This differential characteristic can be extended to $r + i$ rounds, till the bound $p^{-1} \ggg 2^n$ is achieved for the n-bit block cipher.

3.2 Construction of Differential Characteristics using MILP Model

A high probability differential characteristic is required to launch the key recovery attack by adding some rounds on the head and tail of the characteristic. There exists several automated techniques to search the optimal differential characteristics for block ciphers [6]. MILP based technique convert the problem into a linear programming problem and solve it using the optimization problem solvers. MILP models an inequalities based system with the bit variables.
The non-linear function used in a block cipher is the S-box. Therefore, we need to write the all possible input and output differences to the S-box in the linear equations. For this purpose, the difference distribution table (DDT) (Appendix - A) of the S-box is used. Using SageMath [12], we get total 239 inequalities. The constraints of impossible differentials in DDT [9] are used for the reduction by constructing a MILP problem. We have used the Gurobi solver [13] to solve the MILP problem which selects the 21 linear inequalities (Appendix - B) by removing the redundant inequalities. The set of 21 inequalities is used to model the MILP problem to minimize the number of active S-boxes. Further, by analysing the differential distribution probabilities of the S-Box, 1304 inequalities have been generated using SageMath and by applying the reduction procedure we select a set of 20 inequalities (Appendix - C). For the WARP S-Box, there exists three possible probabilities i.e. 1, 2^{-2}, 2^{-3}. Therefore, two extra bits(p_0,p_1) are sufficient to encode the differentials patterns. The differentials patterns, with two extra bits, need to satisfy the Equation 1. These linear inequalities (Appendix - C) are used to model a MILP problem to find the differential characteristic with high probability. In this process, we first minimize the number of S-boxes and then by minimizing the probabilities we get the desired differential characteristics.

$$\begin{aligned}
&(p_0, p_1) = (0,0), if Pr[(x_0, x_1, x_2, x_3) \rightarrow (y_0, y_1, y_2, y_3)] = 1 = 2^{-0} \\
&(p_0, p_1) = (0,1), if Pr[(x_0, x_1, x_2, x_3) \rightarrow (y_0, y_1, y_2, y_3)] = 4/16 = 2^{-2} \\
&(p_0, p_1) = (1,0), if Pr[(x_0, x_1, x_2, x_3) \rightarrow (y_0, y_1, y_2, y_3)] = 2/16 = 2^{-3}
\end{aligned} \tag{1}$$

The objective function to find high probability differential characteristics is to minimize $\sum(3 \times p_0 + 2 \times p_1)$.
To convert the XOR operation (Algorithm 1) into inequalities, for each bit of U and V, we have followed the Equation 2. Here, y refers to the output bit of the XOR operation on the input bits u and v.

$$\begin{aligned}
u + v - y &\geq 0 \\
u - v + y &\geq 0 \\
-u + v + y &\geq 0 \\
u + v + y &\leq 2
\end{aligned} \tag{2}$$

The differential characteristics for the 18-round and 19-round of WARP are described in the following subsections.

3.3 Differential Characteristics for 18-round WARP

Using MILP, a 17-round differential characteristics has been constructed with 57 active S-boxes and 2^{-114} probability. This characteristics is extended by adding one round at the head to get the 18-round characteristic with 61 active S-boxes and 2^{-122} probability as described in Table 4.

Round Index	Input Difference	Probability (p)
Input	0007a000fa7000000a000000d5f000d0	1
1	00700d00a0000000aa00000050000000	2^{-8}
2	0000d50000000000a000000000000a00	2^{-12}
3	000050000000000a000000000000aa00	2^{-16}
4	00000000000000a0000000000000a000	2^{-20}
5	000000000a00000000000000000a0000	2^{-20}
6	0000000aa0000000000a000000a00000	2^{-24}
7	000000aa00000a0000a00a00000a0000	2^{-28}
8	0a0000a00000af000000a00000a00000	2^{-36}
9	a0000f0a0000f000000a00000a000000	2^{-40}
10	0000f0a00000000a00a0000aaf0f0000	2^{-48}
11	00000a000a0000a00f0500aaf0f00a0a	2^{-56}
12	000aaf0aa000000afa500aa00500ada0	2^{-70}
13	00aaf0a000000aaaa000a00a5000df00	2^{-86}
14	00a00500000fa0aa000a00a0000af000	2^{-96}
15	000050000af000a000a500000aa00000	2^{-106}
16	00000000a0000000055000000a000000a	2^{-112}
17	00000a000000000500000000a0000a5	2^{-116}
18	0000a000000a000f0000000fa7000550	2^{-122}

Table 4: 18-round Differential Characteristics (extended from 17-round)

We have also constructed the 18-round differential characteristics without extending the lower round characteristic. Although, the patterns of active S-boxes and differential probabilities are similar to the characteristics described in Table 4. This differential characteristic with 61 active S-boxes and 2^{-122} probability is described in Table 5.

3.4 Differential Characteristics for 19-round WARP

S. Banik [1] has provided a lower bound on the number of active S-boxes for the 19-round WARP. We have constructed a differential characteristic having the same number of active S-boxes as designers bound for 19-round WARP. We get

Round Index	Input Difference	Probability (p)
Input	000af000faf000000a0000005f500050	1
1	00a00500a0000000af000000f0000000	2^{-8}
2	00005f0000000000f000000000000a00	2^{-12}
3	0000f0000000000a000000000000af00	2^{-16}
4	00000000000000a0000000000000f000	2^{-20}
5	000000000f0000000000000000a0000	2^{-20}
6	0000000ff0000000000a000000a00000	2^{-24}
7	000000fa00000a0000a00f00000a0000	2^{-28}
8	0a0000a00000aa000000f00000a00000	2^{-36}
9	a0000f0a0000a000000a00000a000000	2^{-40}
10	0000f0a00000000a00a0000aaa050000	2^{-48}
11	00000a000f0000a00f0d00aaa0500a0a	2^{-56}
12	000aaa0af000000affd00aa00d00ada0	2^{-70}
13	00aaa0a0000005aaf000a00ad000df00	2^{-86}
14	00a00d00000a50aa000a00a0000af000	2^{-96}
15	0000d0000aa000a000ad000005a00000	2^{-106}
16	00000000a00000000dd000005000000a	2^{-112}
17	0000050000000000d00000000a0000ad	2^{-116}
18	00005000000a00070000000da7000dd0	2^{-122}

Table 5: 18-round Differential Characteristics

this characteristic by extending the 18-round differential characteristics (Table 4). We describe the 19-round differential characteristics with 66 active S-boxes and 2^{-132} probability in Table 6.

Round Index	Input Difference	Probability (p)
Input	0007a000fa7000000a000000d5f000d0	1
1	00700d00a0000000aa00000050000000	2^{-8}
2	0000d50000000000a000000000000a00	2^{-12}
3	000050000000000a000000000000aa00	2^{-16}
4	00000000000000a0000000000000a000	2^{-20}
5	000000000a00000000000000000a0000	2^{-20}
6	0000000aa0000000000a000000a00000	2^{-24}
7	000000aa00000a0000a00a00000a0000	2^{-28}
8	0a0000a00000af000000a00000a00000	2^{-36}
9	a0000f0a0000f000000a00000a000000	2^{-40}
10	0000f0a00000000a00a0000aaf0f0000	2^{-48}
11	00000a000a0000a00f0500aaf0f00a0a	2^{-56}
12	000aaf0aa000000afa500aa00500ada0	2^{-70}
13	00aaf0a000000aaaa000a00a5000df00	2^{-86}
14	00a00500000fa0aa000a00a0000af000	2^{-96}
15	000050000af000a000a500000aa00000	2^{-106}
16	00000000a000000005500000a000000a	2^{-112}
17	00000a0000000000500000000a0000a5	2^{-116}
18	0000a000000a000f0000000fa7000550	2^{-122}
19	000f0a000aa500f00d0000fd70005a00	2^{-132}

Table 6: 19-round Differential Characteristics

4 Key Recovery Attack on 21-round WARP

We select the 16-round differential characteristic (round 1 to 17) from the 18-round differential characteristic (Table 5). The probability of the 16-round differential characteristic is 2^{-108}. We add 2 rounds at the beginning and 3 rounds at the end of 16-round differential characteristic as shown in Table 8. Using the 16-round differential characteristic, we can launch a key recovery attack on the 21-round WARP . The 16-round characteristic is chosen in particular because the number of active bits in the head and tail of this characteristic are less. In each round, 64-bit round key is required and it is extracted directly from the 128-bit key K= (K_0, K_1). The key K_0 is used for the odd numbered rounds while the even numbered rounds use the key K_1 (Table 7). We need to guess the round keys which correspond to the actives S-boxes. The round keys used in 1^{st}, 19^{th} and 21^{st} rounds are $(K_0^0, K_0^1, K_0^2, K_0^3, K_0^4, K_0^7, K_0^{10}, K_0^{11}, K_0^{13}, K_0^{14})$ and the keys $(K_1^1, K_1^3, K_1^4, K_1^7, K_1^8, K_1^{10}, K_1^{11}, K_1^{14})$ are used in 2^{nd} and 20^{th} rounds. In total, 72 bits (18 nibbles) of the round keys are used in these rounds.

Round	Key nibbles
1st	$K_0^0, K_0^1, K_0^2, K_0^3, K_0^4, K_0^5, K_0^6, K_0^7, K_0^8, K_0^9, K_0^{10}, K_0^{11}, K_0^{12}, K_0^{13}, K_0^{14}, K_0^{15}$
2nd	$K_1^0, K_1^1, K_1^2, K_1^3, K_1^4, K_1^5, K_1^6, K_1^7, K_1^8, K_1^9, K_1^{10}, K_1^{11}, K_1^{12}, K_1^{13}, K_1^{14}, K_1^{15}$
19th	$K_0^0, K_0^1, K_0^2, K_0^3, K_0^4, K_0^5, K_0^6, K_0^7, K_0^8, K_0^9, K_0^{10}, K_0^{11}, K_0^{12}, K_0^{13}, K_0^{14}, K_0^{15}$
20th	$K_1^0, K_1^1, K_1^2, K_1^3, K_1^4, K_1^5, K_1^6, K_1^7, K_1^8, K_1^9, K_1^{10}, K_1^{11}, K_1^{12}, K_1^{13}, K_1^{14}, K_1^{15}$
21st	$K_0^0, K_0^1, K_0^2, K_0^3, K_0^4, K_0^5, K_0^6, K_0^7, K_0^8, K_0^9, K_0^{10}, K_0^{11}, K_0^{12}, K_0^{13}, K_0^{14}, K_0^{15}$

Table 7: Round Keys of WARP

1st round input = ΔP	00?0af?00????000000000?005??a00?
2nd round input	000af000?a?000000a0000005f?000?0
3rd round input	00a00500a0000000af000000f0000000
.	.
.	.
.	.
19th round input	0000050000000000d00000000a0000ad
20th round input	00005000000a000?0000000?a?000dd0
21st round input	000?0a000?ad00?00?0000???000d?00
21st round output = ΔZ	00??a?0??dd?000???000??a000???00

Table 8: 21-round Differential Attack on WARP

4.1 Data Collection

We can build 2^n structures and each structure traverses the 50 bits (40 undetermined (?) bits and 10 bits with the fixed difference) in ΔP (Table 8). Thus, each structure generates $2^{10} \times 2^{40*2-1} = 2^{89}$ pairs satisfying the differential. Therefore, total number of pairs generated by the 2^n structures are 2^{n+89}.
Such a pair will meet the third round differential in Table 8 with an average probability of 2^{-40}. Then, the probability of obeying the differential after 19th round for the pair encrypted with the right key is 2^{-108}. Therefore, the number of pairs satisfying the differential with a right key guess after 19th round will be $2^{n+89} \times 2^{-40} \times 2^{-108} = 2^{n-59}$. Hence, we choose n=63 so that we could get at least $2^4 = 16$ right pairs under the right key guessing.

4.2 Key Recovery

In this phase, we guess the key bits cosponsoring to the 4-bit key nibbles. This guess includes $K_0^0, K_0^2, K_0^3, K_0^{10}, K_0^{11}, K_0^{13}$ in 1st round, $K_1^3, K_1^7, K_1^{11}, K_1^{14}$ in 2nd round, K_0^0, K_0^3, K_0^{13} in 19th round, $K_1^1, K_1^3, K_1^4, K_1^8, K_1^{10}$ in 20th round and $K_0^1, K_0^4, K_0^7, K_0^{10}, K_0^{11}, K_0^{13}, K_0^{14}$ in 21st round.

Since K_0^0, K_0^3 are involved in 1st and 19th round, K_1^3 is involved in 2nd and 20th round, K_0^{10}, K_0^{11} are involved in 1st and 21st round, and K_0^{13} is involved in 1st, 19th and 21st round. Therefore, total 25-7=18 unique nibbles are involved in the key recovery phase. Hence, we construct $2^{18*4} = 2^{72}$ counters for the possible values of the 72 key bits.

With n=63, we repeat the key guessing procedure for each of the 2^{63+89} pairs. We are left with $2^{63+89-62} = 2^{90}$ pairs after filtered by 62 zero bits in ΔZ. Therefore, the expected counter for a wrong key guess is $2^{63+89-62-40-56} = 2^{-6}$.

4.3 Complexity

With n=63, data complexity of the 21-round differential attack on WARP becomes $2^{63+50} = 2^{113}$. We need to store the counter corresponding to 72 bits of the key, so the memory complexity of the attack becomes 2^{72}. In the first round, we need to guess the 12 key bits corresponding to the three active S-boxes. Therefore, time complexity of the first round becomes $2^{90+12} = 2^{102}$. Similarly, we can calculate the cost of time complexities in the other rounds. Hence, the time complexity of the whole attack is bounded by the 2^{113} chosen plaintexts.

5 Conclusion

In this paper, we have presented a 21-round key recovery attack and the detailed differential characteristics for the 18-round and 19-round WARP. We have achieved the lower bounds, published by the designers, on the number of active S-boxes using MILP-aided search. The differential characteristic for 18 rounds with 61 active S-boxes and probability of 2^{-122} is constructed. The differential characteristic for 19 rounds with 66 active S-boxes and probability of 2^{-132} is also constructed by extending the 18-round characteristic. We have used a 16-round differential characteristic and mounted a key recovery attack on 21-round WARP by adding two rounds on the head and three rounds on the tail of the differential characteristic. The data complexity of the 21-round key recovery differential attack on WARP is 2^{113}. This paper presents the first key recovery attack on 21-round WARP. However, the attack does not pose any threat to the security of full round WARP against differential attack.

References

1. Banik, S., Bao, Z., Isobe, T., Kubo, H., Minematsu, K., Liu, F., Sakamoto, K., Shibata, N., Shigeri, M.: WARP : Revisiting GFN for Lightweight 128-bit Block Cipher. Selected Areas in Cryptography, (2020)
2. Biham, E., Shamir, A.: Differential Cryptanalysis of the full 16-round DES, CRYPTO 92, LNCS, Vol. 740, 487–496, Springer, (1992)
3. Bogdanov A. et al. (2007) PRESENT: An Ultra-Lightweight Block Cipher. In: Paillier P., Verbauwhede I. (eds) Cryptographic Hardware and Embedded Systems - CHES 2007. CHES 2007. Lecture Notes in Computer Science, vol 4727. Springer, Berlin, Heidelberg.

4. Daemen, J., Rijmen, V.: The Design of Rijndael, Springer-Verlag, (2002)
5. Knudsen, L., Robshaw, M.J.B.: Block Cipher Companion, Book Springer, ISBN 978-3-642-17341-7, (2011)
6. Kumar, M., Suresh, TS, Pal, S.K., Panigrahi, A.: Optimal Differential Trails in Lightweight Blokc Ciphers ANU and PICO, Cryptologia, Vol. 44, No. 1, 68–78, (2020)
7. Matsui, M.: On Correlation between the Order of S-boxes and the Strength of DES, EUROCRYPT 94, LNCS, Vol 950, 366–375, Springer, (1994)
8. Mouha, N., Wang, Q., Gu, D., Preneel, B.: Differential and linear cryptanalysis using mixed-integer linear programming. In: Information Security and Cryptology - 7th International Conference, Inscrypt 2011, Beijing, China, November 30 - December 3, 2011. Revised Selected Papers. 57–76, (2011)
9. Sasaki Y., Todo Y. (2017) New Algorithm for Modeling S-box in MILP Based Differential and Division Trail Search. In: Farshim P., Simion E. (eds) Innovative Security Solutions for Information Technology and Communications. SecITC 2017. Lecture Notes in Computer Science, vol 10543. Springer, Cham.
10. Sun S., Hu L., Wang P., Qiao K., Ma X., Song L. (2014) Automatic Security Evaluation and (Related-key) Differential Characteristic Search: Application to SIMON, PRESENT, LBlock, DES(L) and Other Bit-Oriented Block Ciphers. In: Sarkar P., Iwata T. (eds) Advances in Cryptology – ASIACRYPT 2014. ASIACRYPT 2014. Lecture Notes in Computer Science, vol 8873. Springer, Berlin, Heidelberg.
11. Zhu, B., Dong, X., Yu, H.: MILP-Based Differential Attack on Round-Reduced GIFT. In: Topics in Cryptology - CT-RSA 2019, San Francisco, CA, USA, March 4-8, 2019, Proceedings. pp. 372-390, (2019)
12. https://www.sagemath.org/
13. https://www.gurobi.com/
14. https://www.ibm.com/analytics/cplex-optimizer

Appendix

A Difference Distribution Table

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

B Inequalities for Active S-Box Minimization

$$-1*x3-1*x2+0*x1-1*x0+0*y3+0*y2+1*y1+0*y0\geq 2$$
$$-2*x3-1*x2-1*x1-1*x0+1*y3-1*y2+1*y1-1*y0\geq 5$$
$$0*x3+0*x2+1*x1+0*x0-1*y3-1*y2+0*y1-1*y0\geq 2$$
$$0*x3-1*x2-2*x1+2*x0-2*y3+2*y2-1*y1-1*y0\geq 5$$
$$-2*x3-2*x2-1*x1+3*x0-1*y3+3*y2-2*y1-1*y0\geq 6$$
$$0*x3+1*x2+1*x1+1*x0+1*y3-2*y2-1*y1-2*y0\geq 3$$
$$0*x3-1*x2+1*x1-1*x0+0*y3-1*y2+1*y1-1*y0\geq 3$$
$$1*x3+1*x2-1*x1-2*x0-2*y3-2*y2+1*y1+2*y0\geq 5$$
$$0*x3+1*x2-2*x1-2*x0+2*y3+1*y2+1*y1-1*y0\geq 3$$
$$-1*x3+1*x2-2*x1+1*x0+3*y3+1*y2-1*y1+1*y0\geq 1$$
$$-2*x3+3*x2-1*x1-2*x0-1*y3-1*y2-2*y1+3*y0\geq 6$$
$$0*x3-2*x2-2*x1+1*x0+2*y3-1*y2+1*y1+1*y0\geq 3$$
$$3*x3+3*x2+1*x1+2*x0-2*y3+2*y2-2*y1+1*y0\geq 0$$
$$3*x3-2*x2+2*x1+1*x0-1*y3-2*y2-2*y1+1*y0\geq 4$$
$$1*x3-2*x2-1*x1+1*x0-2*y3+2*y2+1*y1-2*y0\geq 5$$
$$1*x3+2*x2+1*x1+2*x0+0*y3-1*y2+0*y1-1*y0\geq 0$$
$$1*x3-2*x2-1*x1-2*x0+2*y3+3*y2+1*y1+3*y0\geq 1$$
$$3*x3+1*x2+2*x1-2*x0-1*y3+1*y2-2*y1-2*y0\geq 4$$
$$-2*x3-1*x2+1*x1-1*x0+3*y3+2*y2+3*y1+2*y0\geq 0$$
$$-1*x3+2*x2-1*x1+2*x0+0*y3+1*y2+2*y1+1*y0\geq 0$$
$$1*x3-1*x2-1*x1-1*x0+0*y3-1*y2-1*y1-1*y0\geq 5$$

C Inequalities for Differential Probability Minimization

$$0*x3+0*x2+0*x1+0*x0+0*y3+0*y2+0*y1+0*y0-1*p0-1*p1\geq 1$$
$$0*x3-1*x2+0*x1-1*x0+0*y3-1*y2+0*y1-1*y0+4*p0+3*p1\geq 0$$
$$0*x3+0*x2+0*x1+0*x0+0*y3+1*y2-1*y1+1*y0+1*p0+0*p1\geq 0$$
$$-1*x3-1*x2+1*x1+2*x0+0*y3+0*y2-1*y1-2*y0+3*p0+4*p1\geq 0$$
$$0*x3-3*x2-2*x1-3*x0+0*y3+1*y2+2*y1+1*y0+6*p0+5*p1\geq 0$$
$$0*x3+2*x2-2*x1-2*x0-3*y3-1*y2-1*y1+2*y0+6*p0+7*p1\geq 0$$
$$7*x3+4*x2+2*x1-2*x0-1*y3+4*y2-5*y1-8*y0+7*p0+10*p1\geq 0$$
$$-4*x3+3*x2-1*x1-2*x0-1*y3-3*y2-2*y1+3*y0+8*p0+10*p1\geq 0$$
$$1*x3+5*x2+2*x1+0*x0+2*y3-1*y2-2*y1-4*y0+2*p0+5*p1\geq 0$$
$$0*x3+1*x2-3*x1+2*x0+1*y3+0*y2-1*y1+2*y0+3*p0+1*p1\geq 0$$
$$-4*x3-2*x2+1*x1-2*x0+2*y3+1*y2+5*y1+1*y0+1*p0+4*p1\geq 0$$
$$0*x3+2*x2+3*x1-1*x0-2*y3-1*y2+0*y1-1*y0+1*p0+4*p1\geq 0$$
$$0*x3+1*x2-1*x1+1*x0+1*y3-1*y2+1*y1-1*y0+3*p0+1*p1\geq 0$$
$$7*x3-2*x2+2*x1+4*x0-1*y3-8*y2-5*y1+4*y0+7*p0+10*p1\geq 0$$
$$2*x3+1*x2+5*x1+1*x0-4*y3-2*y2+1*y1-2*y0+1*p0+4*p1\geq 0$$
$$1*x3+2*x2+0*x1+2*x0+1*y3+2*y2+0*y1+2*y0-2*p0-3*p1\geq 0$$
$$-2*x3+2*x2-4*x1-4*x0+5*y3+2*y2+1*y1-1*y0+5*p0+8*p1\geq 0$$
$$0*x3+1*x2-3*x1+2*x0-1*y3+2*y2-1*y1-1*y0+5*p0+3*p1\geq 0$$
$$-2*x3-4*x2-1*x1+2*x0-1*y3+2*y2+3*y1-1*y0+3*p0+8*p1\geq 0$$
$$2*x3-4*x2-2*x1-1*x0+1*y3-1*y2-2*y1+2*y0+6*p0+10*p1\geq 0$$