The SAT-Based Automatic Searching and Experimental Verification for Differential Characteristics with Application to Midori64

Yingying Li and Qichun Wang(✉)

School of Computer and Electronic Information,
Nanjing Normal University, Nanjing, China
yyli@nnu.edu.cn, qcwang@fudan.edu.cn

Abstract. In this paper, we show that it is inaccurate to apply the hypothesis of independent round keys to search for differential characteristics of a block cipher with a simple key schedule. Therefore, the derived differential characteristics may be invalid. We develop a SAT-based algorithm to verify the validity of differential characteristics. Furthermore, we take the key schedule into account and thus put forward an algorithm to directly find the valid differential characteristics. All experiments are performed on Midori64 and we find some interesting results.

Keywords: Lightweight block cipher · Differential characteristic · SAT · Midori64 · Hypothesis of independent round keys.

1 Introduction

Midori [2] presented at ASIACRYPT 2015 is a family of lightweight block ciphers with low energy consumption. The family is composed of two versions Midori64 and Midori128, which encrypt 64-bit and 128-bit plaintexts, respectively. Due to the small state space of Midori64, we focus on Midori64 in this paper.

The most critical step of differential cryptanalysis is to obtain the differential characteristics with high probability. In general, automatic search methods based on MILP and SAT/SMT are utilized to find them under the hypothesis of independent round keys [1, 3, 8]. However, for Midori64, its round keys are not independent because of its simple key schedule. Furthermore, there are no right pairs that follow the expected propagation of the differential characteristic [4]. That is, the differential characteristic is invalid.

This inspires us to develop an accurate SAT-based method for verifying the validity of the differential characteristics and encourages us to improve the existing algorithms for directly finding the valid differential characteristics. Our main contributions are listed in the following:

- Using the SAT-based method under the hypothesis of independent round keys, see Algorithm 1, we obtain the upper bounds on the probability of the best differential characteristics for full-round Midori64.

- We propose a SAT-based method to verify the validity of a differential characteristic, see Algorithm 2. For Midori64, we apply this method to test whether the differential characteristics obtained by Algorithm 1 are valid.
- In knowing the upper bounds on the probability of the best differential characteristics, we take the key schedule into account and thus put forward an algorithm to directly search for valid characteristics of a block cipher, see Algorithm 3. As a result, we improve some previous results.

The rest of this paper is organized as follows. In Section 2, we give a brief description of Midori64. In Section 3, the SAT-based method under the hypothesis of independent round keys is used for Midori64. In Section 4, we present an accurate SAT-based algorithm to verify the validity of differential characteristics. In Section 5, we give the algorithm directly finding the valid differential characteristics with application to Midori64. In Section 6, we conclude this paper.

2 A Brief Description of Midori64

Midori64 has a SPN structure, whose state size is 64 bits, key size is 128 bits and round number R is 16.

Each round function of Midori64 is composed of the following four operations. SubCell (SC) is the only nonlinear operation where 16 4-bit S-boxes are applied to each nibble of the state in parallel. ShuffleCell (SFC) applies a nibble-wise permutation to the state. MixColumn (MC) performs a linear transformation on each 4-nibble column of the state. KeyAdd (KA) uses a XOR operation, which bitwise XORs the i-th 64-bit round key RK_i to each bit of the state.

The data encryption process of Midori64 is as follows: Firstly, using KA, a 64-bit whitening key WK is XORed to each bit of the state. Then, the round function is performed $R-1$ times. Finally, SC is executed, and again KA using WK is carried out.

The key schedule of Midori64 is relatively simple and uses a 128-bit master key K that is composed of two 64-bit keys K_0 and K_1: $K = K_0||K_1$. The whitening key WK is computed as $WK = K_0 \oplus K_1$ and the round key is $RK_i = K_{i \bmod 2} \oplus \alpha_i$, $0 \leq i \leq 14$, where α_i is the round constant. More details about Midori64 are depicted in design documentation [2].

3 The Method Proposed by Sun et al. with Application to Midori64

In this section, we apply the SAT-based method proposed by Sun et al. to find the upper bounds on the probability of the best differential characteristics for full-round Midori64. We use the SAT solver called Cryptominisat [6] to do our work. It accepts CNF (Conjunctive Normal Form) files as the standard input, which is equivalent to the product-of-sum representation of Boolean functions.

In the following, we give a general framework of the SAT-based method proposed by Sun et al. [8], see Algorithm 1. However, we emphasize that this

approach is based on the hypothesis of independent round keys. If this hypothesis of a block cipher is weak, the derived differential characteristics may be invalid. Thus, the probability of the best differential characteristics is only the upper bound. It depends on whether the derived characteristics are valid. If valid, the bound is tight; otherwise, is not tight.

Algorithm 1 The SAT-based search algorithm under the hypothesis of independent round keys

Require: the total round R
Ensure: the upper bounds *bound* on the best probability for a R-round primitive
1: *bound* ← list([0,0,$\cdots$]) ▷ store R-round information
2: *result* ← - 1 ▷ the weight of the best probability
3: **for** $r \leftarrow 0$ to $R-1$ **do**
4: *flag* ← false
5: **while** *flag* is false **do**
6: *result* ← *result* + 1
7: *model1* ← ()
8: *model1* ← BuildModel1(r, *model1*, *result*,*bound*)
9: *Flag* ← the result obtained by solving the *model1*
10: **if** *Flag* is "SAT" **then**
11: *flag* ← true
12: **end if**
13: **end while**
14: *bound*[r] ← *result*
15: **end for**
16: **return** *bound*
17:
18: **function** BuildModel1(r, *model1*, *result*, *bound*)
19: **for** $i \leftarrow 0$ to r - 1 **do**
20: *model1* += the differential propagation rules for the i-th round primitive
21: **end for**
22: *model1* += the model of objective function about the weight *result*
23: *model1* += the model of Matsui's bounding conditions created with *bound*
24: **return** *model1*
25: **end function**

In lines 7-9 of Algorithm 1, the process of searching with the SAT solver can be summarized as follows: Firstly, the search problem is expressed as a set of CNF clauses, and thus the SAT model is established. Then, the model is solved by the solver. Finally, if there is a solution, then the solver returns "SAT" and a solution is extracted; otherwise, returns "UNSAT".

For different block ciphers, BUILDMODEL1() is the only different part. To apply Algorithm 1 to Midori64, we need to establish the differential propagation models for all the operations that include XOR, S-box, SFC, and MC.

For bitwise XOR operation $\alpha_0 \oplus \alpha_1 \oplus \cdots \oplus \alpha_{n-1} = \beta$, we define a $(n+1)$-bit Boolean function $f(\alpha_0||\alpha_1||\cdots||\alpha_{n-1}||\beta)$ as

$$f(\alpha_0||\alpha_1||\cdots||\alpha_{n-1}||\beta) = \begin{cases} 1, \ if \ \alpha_0 \oplus \alpha_1 \oplus \cdots \oplus \alpha_{n-1} = \beta \\ 0, \ else \end{cases}.$$

For 4-bit S-box operation, let $x \in F_2^4$ and $y \in F_2^4$ be the input and output differences of differential distribution table (DDT) of S-box, respectively. And p is the probability of a differential propagation in the DDT. We introduce 3 extra binary variables w_0, w_1, w_2 to encode the weight of probability, as follows:

$$w = (w_0, w_1, w_2) = \begin{cases} (0,0,0), \ if \ p = 2^{-0} \\ (0,1,1), \ if \ p = 2^{-2} \\ (1,1,1), \ if \ p = 2^{-3} \end{cases}.$$

We define a 11-bit Boolean function $f(x||y||w)$ as

$$f(x||y||w) = \begin{cases} 1, \ if \ x \to y \ is \ a \ possible \ propagation \ with - log_2 \ p = \sum_{k=0}^{2} w_k \\ 0, \ else \end{cases}.$$

For SFC operation, we only need to change the positions of bits, which indicates that the extra CNF clauses are not required.

For MC operation, we can find its primitive representation [7]. Thus, the 4×4 involutive matrix over field F_2^4 can be converted to a 16×16 binary matrix. And the MC operation is converted to 64 XOR operations.

Use the software Logic Friday [5] to obtain the minimum product-of-sum representations of all operations and thus generate a set of smaller CNF clauses.

Our goal is the r-round upper bounds of probability. Express the extra variable of the j-th S-box in the i-th round as $w_k^{(i,j)}$, where $0 \le i \le r-1$, $0 \le j \le 15$, and $0 \le k \le 2$. Thus, the objective function is expressed as $\sum_{i=0}^{r-1} \sum_{j=0}^{15} \sum_{k=0}^{2} w_k^{(i,j)}$. It can be abstracted as the Boolean cardinality constraint $\sum_{i=0}^{n-1} x_i \le z$, where z is a non-negative integer. This requires the solver to find such a differential characteristic that the weight of differential probability is less than or equal to z. For more information about the modeling of this constraint, see [8].

Matsui's bounding conditions are encoded to the SAT model for accelerating the search. These conditions take full advantage of the fact that the upper bounds on the probability of short characteristics are known. Similarly, for more information about the modeling of those conditions, see [8].

Table 1. The weight of the upper bounds on the probability of the best differential characteristics for full-round Midori64.

Round	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$-log_2\ p$	2	8	14	32	46	60	70	76	82	100	114	124	134	144	150

Thus, the involved SAT model has been completed. Using Algorithm 1, we find the upper bounds on the probability of the best differential characteristics for full-round Midori64, as shown in Table 1. It more accurately evaluates the security of Midori64 against single-key differential cryptanalysis, which is roughly

estimated by the low bounds on the number of differential active S-boxes in the design document of Midori64 [2]. Notice the fact that the block size for Midori64 is 64 bits. From Table 1, 7-round Midori64 is sufficient to resist single-key differential cryptanalysis, because $2^{-70} \leq 2^{-64} \leq 2^{-60}$.

4 Verifying the Validity of Differential Characteristics Based on SAT

In this section, we show that the derived differential characteristics under the hypothesis of independent round keys may be invalid because round keys of Midori64 are not independent. As the work in [4], we only focus on whether a differential characteristic is valid and ignore the value of its non-zero probability. If the characteristic Q with non-zero probability is invalid, the work based on Q cannot reflect the security of a block cipher against differential cryptanalysis.

Algorithm 2 The SAT-based algorithm to verify the validity of a differential characteristic

Require: a r-round differential characteristic Q
Ensure: The validity of Q

model2 ← ()
model2 ← BuildModel2(r, *model2*)
Flag ← the result obtained by solving the *model2*
if *Flag* is "SAT" **then**
    solution ← a valid key and the corresponding right pair following Q
    return ["SAT", *solution*] ▷ Q is valid
else
    return "UNSAT" ▷ Q is invalid
end if

function BuildModel2(r, *model2*)
    for $i \leftarrow 0$ to r - 1 **do**
        model2 += the constraint rules of characteristic Q on intermediate states of a pair of plaintexts in the i-th round
        model2 += the value propagation rules of the encryption part for a pair of plaintexts in the i-th round
        model2 += the value propagation rules of the key schedule part for a pair of plaintexts in the i-th round
        model2 += the rules of linking both parts via the round key k_i
    end for
    return *model2*
end function

Therefore, we present an accurate SAT-based algorithm to verify the validity of a differential characteristic, see Algorithm 2. It encrypts separately a pair of plaintexts with a key for the primitive and thus it can be used to check whether the XOR value of two plaintexts in each round satisfies the difference value in each round. If the SAT model has a solution, then a valid key and

the corresponding right pair following the differential characteristic are output; otherwise, the differential characteristic will be invalid, which also indicates that there are no valid keys that follow the propagation of Q.

Next, we give a specific description of BUILDMODEL2() for Midori64. We need to establish the value propagation models for all operations. Then, according to the structure of the block cipher, the model of each operation is connected to establish the r-round propagation model of a pair of plaintexts. XOR, SFC, and MC of all operations are linear and can be modeled similarly to the corresponding differential propagations in Section 3. Here, we introduce the SAT model for the value propagations of the non-linear operation S-box.

For 4-bit S-box operation $y = S(x)$, where $x \in F_2^4$ and $y \in F_2^4$ are the input and the output values of S-box, respectively. We define a 8-bit Boolean function $f(x||y)$ as

$$f(x||y) = \begin{cases} 1, \ if\ x \to y\ is\ a\ possible\ propagation\ with\ y = S(x) \\ 0, \ else \end{cases}.$$

Similarly, use Logic Friday to generate a set of smaller CNF clauses. So far, the SAT model of each operation of Midori64 has been completed. Thus, we can use Algorithm 2 to verify some of the characteristics obtained by Algorithm 1.

We modified Algorithm 1 to output multiple differential characteristics of r-round Midori64, where $1 \leq r \leq 6$. Usually, the solver only outputs one solution. To find multiple solutions, we utilize its incremental property, which allows the solver to record the current information. After the solver outputs a solution, an additional CNF clause is added to the SAT model to prohibit this solution. Then, the solver is asked to give a solution again, and so on, until the solver returns "UNSAT". Specifically, for a n-bit variable $(x_0, x_1, \cdots, x_{n-1})$ with its specific solution $(k_0, k_1, \cdots, k_{n-1})$, the CNF clause $\bigvee_{i=0}^{n-1} (x_i \oplus k_i) = 1$ is appended.

We apply Algorithm 2 to verify the validity of these differential characteristics. The results show that some of them are valid, which indicates that the upper bounds of $1 \leq r \leq 6$ rounds obtained by Algorithm 1 are tight. However, some of them are invalid, which also indicates that the hypothesis of independent round keys is inaccurate for Midori64.

Such experimental results remind us that we can use this hypothesis to roughly assess the resistance of a block cipher against differential cryptanalysis. However, when we want to obtain specific differential characteristics for differential attacks, we should pay attention to the validity of characteristics.

5 Our New Algorithm for Finding Valid Differential Characteristics

The first two sections explain that the derived differential characteristics under the hypothesis of independent round keys may be invalid. In the following, we build a SAT model that involves both the differential and value propagations of a primitive to directly search for valid differential characteristics.

In knowing the upper bounds on the best probability of the r-round primitive, we take the key schedule into account and thus propose an improved search method for directly finding a valid characteristic, see Algorithm 3.

Algorithm 3 The improved SAT-based search algorithm for directly finding a valid r-round characteristic

Require: the target round r
Ensure: a valid r-round differential characteristic

```
1: bound ← list(R)   ▷ store the known upper bounds on the probability of R rounds
2: result ← bound[r]   ▷ the weight of the best r-round probability
3: MAX_WEIGHT ← 10000
4: while result < MAX_WEIGHT do
5:     model3 ← ()
6:     model3 ← BuildModel3(model3, result, bound)
7:     Flag ← the result obtained by solving the model3
8:     if Flag is "SAT" then
9:         solution ← a valid r-round characteristic
10:        return ["SAT", solution]
11:    end if
12:    result ← result + 1
13: end while
14:
15: function BuildModel3(model3, result, bound)
16:    BuildModel1(r, model3, result, bound)   ▷ the differential propagations
17:    BuildModel2(r, model3)   ▷ the value propagations
18:    return model3
19: end function
```

To improve efficiency of the search, we use the upper bounds obtained by Algorithm 1 to avoid the search in the probability space for which no characteristics exist, as shown in the 1-th row of Algorithm 3. The BUILDMODEL3() part not only searches for a characteristic but also verifies its validity. Similarly, incremental property of the solver can be used to obtain multiple valid characteristics.

The greater probability of two 5-round characteristics was 2^{-52} in [9]. Using our Algorithm 3, we search for the best valid differential characteristics of Midori64. We find some 5-round characteristics with probability 2^{-46}, which increases a factor of 2^6 than the probability 2^{-52}. Furthermore, we also find some 6-round characteristics with probability 2^{-60}, which means that we may attack Midori64 with one more round than the result of [9].

These multiple valid characteristics may be used to launch better differential attacks than the existing ones. And for Midori64, the fewer active nibbles of the input and output differences of a characteristic, the more conducive to a differential key recovery attack. Note that we do not consider this factor when searching for characteristics. Thus, we can continue the study of selecting advantageous ones among these multiple valid characteristics.

6 Conclusion and Future Work

In this paper, we show by experimentation that the derived characteristics for Midori64 under the hypothesis of independent round keys may be invalid. Furthermore, we propose a new algorithm to directly search for valid characteristics. Using it, we obtain some better valid characteristics, which may improve the complexity of existing key recovery attacks of Midori64.

In the future, on the one hand, we can search for advantageous characteristics to perform better differential key recovery attacks on Midori64. On the other hand, we need to be careful in presuming that the hypothesis of independent round keys applies to a block cipher.

Acknowledgement

We would like to thank the anonymous reviewers for their helpful comments. This work was supported by National Natural Science Foundation of China (No. 62172230), National Natural Science Foundation of Jiangsu Province (No. BK20201369) and Open Research Program of Shanghai Key Lab of Intelligent Information Processing (No. IIPL201901).

References

1. Ankele, R., Kölbl, S.: Mind the gap - a closer look at the security of block ciphers against differential cryptanalysis. In: Cid, C., Jacobson Jr., M.J. (eds.) Selected Areas in Cryptography – SAC 2018. pp. 163–190. Springer (2019). https://doi.org/10.1007/978-3-030-10970-7_8
2. Banik, S., Bogdanov, A., Isobe, T., Shibutani, K., Hiwatari, H., Akishita, T., Regazzoni, F.: Midori: a block cipher for low energy. In: Iwata, T., Cheon, J.H. (eds.) Advances in Cryptology – ASIACRYPT 2015. pp. 411–436. Springer (2015). https://doi.org/10.1007/978-3-662-48800-3_17
3. Lai, X., Massey, J.L., Murphy, S.: Markov ciphers and differential cryptanalysis. In: Davies, D.W. (ed.) Advances in Cryptology – EUROCRYPT 1991. pp. 17–38. Springer (1991). https://doi.org/10.1007/3-540-46416-6_2
4. Liu, Y., Zhang, W., Sun, B., Rijmen, V., Liu, G., Li, C., Fu, S., Cao, M.: The phantom of differential characteristics. Designs, Codes and Cryptography **88**(11), 2289–2311 (2020)
5. Rickmann, S.: Logic friday (version 1.1.3) [computer software] (2011)
6. Soos, M.: Cryptominisat SAT solver (2009), https://github.com/msoos/cryptominisat
7. Sun, B., Liu, Z., Rijmen, V., Li, R., Cheng, L., Wang, Q., Alkhzaimi, H., Li, C.: Links among impossible differential, integral and zero correlation linear cryptanalysis. In: Gennaro, R., Robshaw, M. (eds.) Advances in Cryptology – CRYPTO 2015. pp. 95–115. Springer (2015). https://doi.org/10.1007/978-3-662-47989-6_5
8. Sun, L., Wang, W., Wang, M.: Accelerating the search of differential and linear characteristics with the SAT method. IACR Transactions on Symmetric Cryptology **2021**(1), 269–315 (2021)
9. Zhao, H., Han, G., Wang, L., Wang, W.: MILP-based differential cryptanalysis on round-reduced Midori64. IEEE Access **8**, 95888–95896 (2020)