New Automatic Search Tool for Impossible Differentials and Zero-Correlation Linear Approximations

Tingting Cui[1], Keting Jia[2,3], Kai Fu[4], Shiyao Chen[1], Meiqin Wang[1,3]

[1] Key Laboratory of Cryptologic Technology and Information Security, Ministry of Education, Shandong University, Jinan 250100, China
[2] Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China
[3] State Key Laboratory of Cryptology, P.O. Box 5159, Beijing 100878, China
[4] China Academy of Information and Communications Technology

Abstract. Impossible differential cryptanalysis and zero-correlation linear cryptanalysis are two of the most useful cryptanalysis methods in the field of symmetric ciphers. Until now, there are several automatic search tools for impossible differentials such as $\mathcal{U}$-method and UID-method, which are all independent of the non-linear S-boxes. Since the differential and linear properties can also contribute to the search of impossible differentials and zero-correlation linear approximations respectively, it is meaningful to study the search with considering the properties of non-linear components. In this paper, we propose an automatic search tool for impossible differentials and zero-correlation linear approximations in both ARX ciphers and ciphers with S-box, which is the first widely applicable one that considers the influence of non-linear operations, especially in ARX ciphers. What's more, this tool can be used to prove whether there are impossible differentials (zero-correlation linear approximations) in certain rounds of a target cipher, particularly for certain subset of input and output differences (masks) patterns. As applications, we use this automatic tool on HIGHT and LBlock ciphers. Consequently, we find total 4 impossible differentials and 4 zero-correlation linear approximations for 17-round HIGHT which are the longest ones until now, and find six 16-round related-key impossible differentials for LBlock, which are the best ones up to now.

Keywords: Automatic search tool, (related-key) impossible differential, zero-correlation linear approximation, HIGHT, LBlock

1 Introduction

Impossible differential cryptanalysis (IDC), introduced by Biham *et al.* and Knudsen to attack Skipjack in [2] and DEAL [20] respectively, unlike the differential cryptanalysis [3] which aims to find a differential characteristic with high probability, tries to find the longest impossible differential, i.e., to find the longest differential with probability 0. It is also a powerful cryptanalysis tool.

Since it was proposed, impossible differential cryptanalysis has been used to lots of block ciphers such as AES [24], LBlock [12], Camellia [6, 12] and so on. As the counterpart of impossible differential cryptanalysis, zero-correlation linear cryptanalysis, a variant of linear cryptanalysis [25], was proposed by Bogdanov *et al.* in [7] and improved in [8–11]. Similar to the idea of impossible differential, its purpose is to find a linear approximation with probability exactly 1/2. In [29], Sun *et al.* proposed that in some cases, a zero-correlation linear approximation was equivalent to an impossible differential.

How to find the best impossible differential for a target cipher is a point of focus in the field of systemic ciphers. Until now, for the automatic search of impossible differentials, several approaches have been proposed such as $\mathcal{U}$-method [19], UID-method [23] and the extended tool of them generalized by Wu and Wang in [38] [5]. Moreover, it has been proved that WW-method could find all truncated impossible differentials of a word-orient block cipher which are independent of the non-linear components such as S-box. However, the differential property of S-box is wasted in above-mentioned methods. At Crypto 2016, an new impossible differential search method by Derbez and Fouque [14] was proposed, which involved the key-recovery phase into the search and could directly calculate the time and memory complexities, but it still ignore the differential property of S-box. What's more, in ARX ciphers there are none of widely applicable automatic approaches because of the modular addition operation. If we can exploit the differential property of non-linear parts in the search of impossible differentials, it might be more accurate to evaluate the security of target block ciphers and more possible to find longer impossible differentials. Inspired by the automatic search of differentials and linear approximations with MILP method introduced by [30, 15], we hope to search the impossible differentials and zero-correlation linear approximations [6] with MILP models as well.

Mixed Integer Linear Programming (MILP) problem is a mathematical optimization problem in which only some variables are constrained to be integers and the goal is to find the minimum or maximum of the objective function, for instance, covering problem and packing problem. It was introduced into differential and linear cryptanalysis by Mouha *et al.* and Wu *et al.* in [26] and [37] respectively, later improved in [31, 30, 15, 32]. According to its applications on the search of differentials and linear approximations in block ciphers, every operation in a certain cipher can be exactly described with inequalities system including non-linear operations such as S-box and modular addition. By exploiting mathematical optimization software which can expedite the feasible and optimized solution, we can search the optimal characteristic for the target cipher with suitable executable time. Can MILP method be used to the search of impossible differentials and ZC approximations with suitable time? This is the motivation for us to do this work.

[5] This method is renamed as WW-method through this paper.

[6] For the sake of simplicity, zero-correlation linear approximation is renamed as ZC approximation through this paper.

1.1 Contributions

Propose an automatic search tool for impossible differentials and ZC approximations in both ARX ciphers and ciphers with S-box. Impossible differential cryptanalysis and zero-correlation linear cryptanalysis are two efficient cryptanalysis methods in the field of symmetric ciphers. Up to now, several existed automatic search approaches such as $\mathcal{U}$-method, UID-method and WW-method, have all been independent of the non-linear components such as S-box and modulo addition. In fact, differential and linear properties of non-linear operations are conducive to the search of longer impossible differentials and ZC approximations. In this paper, combined the properties of non-linear operations, we propose a new automatic search tool for ARX ciphers and ciphers with S-box based on [15] and [30]. With this new tool, all operations are considered including the non-linear operations, so we can not only find the previous results, but also may find longer impossible differentials and ZC approximations. In addition, by traversing a special subset of input and output differences depending on the actual cipher, this method can prove whether there is an impossible differential or ZC approximation in certain rounds of target cipher or not in this subset. As far as we know, this method is the first automatic search tool which takes the properties of non-linear components into consideration for both ARX ciphers and ciphers with S-box. With this tool, it is more likely to find the best impossible differential or ZC approximation.

Application to HIGHT Cipher. HIGHT cipher, introduced by Hong *et al.* at CHES 2006 [16], is an ISO standard lightweight block cipher. Its block size and key size are 64 bits and 128 bits respectively, and it totally has 32 rounds. The longest previous impossible differential and ZC approximation are both 16 rounds, which are introduced in [22, 13, 28] and by Wen *et al.* in [34] respectively. In our work, we use the proposed automatic tool to search all cases of 17-round impossible differentials (ZC approximations) that both hamming weights of input and output differences (masks) are one. As a result, we totally find 4 impossible differentials and 4 ZC approximations for 17-round HIGHT, which are the longest ones until now. The results of impossible differentials and ZC approximations on HIGHT are summarized in Table 1.

Table 1. Summary of impossible differentials and ZC approximations on HIGHT

Type	Round	Resource
Impossible differential	16	[22]
Impossible differential	16	[13]
Impossible differential	16	[28]
Impossible differential	**17**	**Sec. 4.2**
ZC approximation	16	[34]
ZC approximation	**17**	**Sec.4.3**

Application to LBlock Cipher. LBlock cipher, designed by Wu and Zhang in [39], is an efficient lightweight block cipher. Its block size and key size are 64 bits and 80 bits. It applies a 32-round modified Feistel structure. Under the related-key setting, Minier and Naya-Plasencia found a 15-round related-key impossible differential in [36], then Wen *et al.* found two 16-round related-key impossible differentials in [35]. But Wen *et al.*'s two differentials are right only under part of master key pairs which satisfy one of the given two key differences. With our new search tool, we build a MILP model for LBlock and only search the cases that the difference of master key has only one nonzero bit and the input and output differences both have no more than one nonzero bit. In the end, we find six 16-round related-key impossible differentials. As long as the master key pair satisfies one of the given differences, such related-key impossible differential is right in our work. The results of related-key impossible differentials for LBlock are summarized in Table 2.

Table 2. Summary of related-key impossible differentials on LBlock

Type	Round	Number of Keys	Resource
Related-key imp. diff.	15	2	[36]
Related-key imp. diff. *	16	4	[35]
Related-key imp. diff.	**16**	**2**	**Sec.5.2**

[1] Related-key imp. diff.: Related-key impossible differential.
[2] *: This related-key impossible differential is right only for part of master key pairs which satisfied the given difference of master key. When such related-key impossible differentials are used to attack target cipher, it is necessary to use four master keys (two related-key impossible differentials).

1.2 Outline

This paper is organized as follows. In section 2, we propose an automatic tool for search of impossible differentials and ZC approximations in both ARX ciphers and ciphers with S-box. Then in section 3, a verification algorithm is presented. As applications, we use this tool to search longer impossible differentials and ZC approximations for HIGHT in section 4 and improved related-key impossible differentials for LBlock in section 5. Finally, section 6 concludes this paper.

2 Automatic Tool for Search of Impossible Differentials and ZC Approximations

In this section, we propose an automatic tool for search of impossible differentials in both ARX ciphers and ciphers with S-box. Like the idea of MILP models for differential cryptanalysis in previous work, we firstly utilize linear inequalities to

exactly describe every component in target cipher as well. But we are indifferent to the objective function, and only interested in whether there is a solution for the whole inequalities system with fixed input and output differences or not. If not, the fixed input and output differences can lead to an impossible differential, which is expected. In section 2.1 and 2.2, we will build the models for search of impossible differentials in ARX ciphers and ciphers with S-box respectively.

2.1 Impossible Differential Model for ARX Ciphers

ARX ciphers are designed by combining modular addition, bit rotation and XOR operations. For each operation, there is a set of inequalities to exactly depict it.

Constraints for XOR and Bit Rotation XOR and bit rotation are both the linear operations. For every XOR operation with bit-level input and output differences a, b and c, the constraints below can perfectly describe it, according to Sun *et al.*'s work in [30].

$$\begin{aligned} a+b+c &\leq 2 \\ a+b+c &\geq 2d_{\oplus} \\ d_{\oplus} \geq a, d_{\oplus} &\geq b, d_{\oplus} \geq c \end{aligned} \tag{1}$$

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

Actually we can simply use one equation below to exactly describe the XOR operation, because all variables in the model are $0-1$ variables.

$$a+b+c=2d_{\oplus} \tag{2}$$

For the case of circular shift, since it only transforms the position of its input bits, so we can easily build linear equations for the related bits.

Constraints for Modular Addition In [21], Lipmaa and Moriai proposed a method to verify whether a given differential characteristic is possible or not. For sake of simplicity, Fu *et al.* summarized this method into a theorem in [15] as follows:

Theorem 1 (see [21, 15]). *The differential* $(\alpha, \beta \rightarrow \gamma)$ *satisfies* $\gamma = \alpha + \beta$ *iff* $(\alpha[0] \oplus \beta[0] \oplus \gamma[0]) = 0$ *and* $\alpha[i-1] = \beta[i-1] = \gamma[i-1] = \alpha[i] \oplus \beta[i] \oplus \gamma[i]$ *when* $\alpha[i-1] = \beta[i-1] = \gamma[i-1]$, $i \in [1, n-1]$.

In order to describe the first condition $\alpha[0] \oplus \beta[0] \oplus \gamma[0] = 0$ in Theorem 1, we can utilize one equality to satisfy it as follows:

$$\alpha[0] + \beta[0] + \gamma[0] = 2d_{\oplus} \tag{3}$$

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

When $i \in [1, n-1]$, there are 56 possible patterns for $(\alpha[i], \beta[i], \gamma[i],\ \alpha[i+1], \beta[i+1], \gamma[i+1], \neg eq(\alpha[i], \beta[i], \gamma[i]))$ to meet the second condition in Theorem 1, where $\neg eq(\alpha[i], \beta[i], \gamma[i]) = 1$, if $\alpha[i] = \beta[i] = \gamma[i]$, otherwise, is zero. In [15], Fu *et al.* used 13 inequalities to exactly describe these 56 possible patterns for each $i \in [1, n-1]$ as follows.

$$
\begin{array}{rl}
\beta[i] & -\gamma[i] + (\neg eq(\alpha[i], \beta[i], \gamma[i])) \geq 0, \\
\alpha[i] & -\beta[i] + (\neg eq(\alpha[i], \beta[i], \gamma[i])) \geq 0, \\
-\alpha[i] & +\gamma[i] + (\neg eq(\alpha[i], \beta[i], \gamma[i])) \geq 0, \\
-\alpha[i] & -\beta[i] - \gamma[i] - (\neg eq(\alpha[i], \beta[i], \gamma[i])) \geq -3, \\
\alpha[i] & +\beta[i] + \gamma[i] - (\neg eq(\alpha[i], \beta[i], \gamma[i])) \geq 0, \\
-\beta[i] & +\alpha[i+1] + \beta[i+1] + \gamma[i+1] + (\neg eq(\alpha[i], \beta[i], \gamma[i])) \geq 0, \\
\beta[i] & +\alpha[i+1] - \beta[i+1] + \gamma[i+1] + (\neg eq(\alpha[i], \beta[i], \gamma[i])) \geq 0, \\
\beta[i] & -\alpha[i+1] + \beta[i+1] + \gamma[i+1] + (\neg eq(\alpha[i], \beta[i], \gamma[i])) \geq 0, \\
\alpha[i] & +\alpha[i+1] + \beta[i+1] - \gamma[i+1] + (\neg eq(\alpha[i], \beta[i], \gamma[i])) \geq 0, \\
\gamma[i] & -\alpha[i+1] - \beta[i+1] - \gamma[i+1] + (\neg eq(\alpha[i], \beta[i], \gamma[i])) \geq -2, \\
-\beta[i] & +\alpha[i+1] - \beta[i+1] - \gamma[i+1] + (\neg eq(\alpha[i], \beta[i], \gamma[i])) \geq -2, \\
-\beta[i] & -\alpha[i+1] + \beta[i+1] - \gamma[i+1] + (\neg eq(\alpha[i], \beta[i], \gamma[i])) \geq -2, \\
-\beta[i] & -\alpha[i+1] - \beta[i+1] + \gamma[i+1] + (\neg eq(\alpha[i], \beta[i], \gamma[i])) \geq -2.
\end{array}
\tag{4}
$$

Note that this model for differential characteristic of modular addtion is suitable for cases that only two independent inputs are involved.

Up to now, every operation in ARX cipher has been exactly described with a set of inequalities. By combining all inequalities for every operation in the target ARX cipher, the whole inequalities system can perfectly describe this cipher, and each solution is a differential characteristic. When we fixed the input and output differences, if the inequalities system is infeasible, it means this is an impossible differential. By traversing a special subset of (input, output) differences in the MILP model, we confirm whether there exists an impossible differential or not for a certain reduced-round ARX cipher in this subset. Usually, this subset is decided according to the feature of the given cipher. Without loss of generality, we denote such subset as $(\Delta \to \Gamma)$, where Δ and Γ are the sets of possible input and output differences respectively. In Algorithm 1, we explain how to implement the search of impossible differentials with Gurobi Optimization, when the MILP model file "model.lp" is already produced just as those in [32, 15].

2.2 Impossible Differential Model for Ciphers with S-box

Comparing with ARX ciphers, lots of block ciphers use S-box layer as the non-linear operations rather than modular addition, and linear operations maybe more complicated with combining many XOR, rotation operations and simple permutations. For the sake of simplicity, we don't depict the linear operations in detail as they have been exactly described in section 2.1.

Algorithm 1: General search process for impossible differentials

```
   // Assume the block size is n.
 1 def mycallback(model,where):
 2       if where == GRB.Callback.MIP:
 3         best = model.cbGet(GRB.Callback.MIP_OBJBST)
 4       if best ≥ 0:
 5         m.terminate()
   // mycallback function is to terminate the optimization if a
      solution is already appeared.
 6 for All input differences Δx_i ∈ Δ do
 7     for All output differences Δy_j ∈ Γ do
 8         if i = 0 and j = 0 then
 9             Add all constraints about the fixed input and output differences into
                "model.lp";
10         else
11             Change all constraints about the fixed input and output differences
                in "model.lp";
12         m=read('model.lp');
13         m.optimize(mycallback);
14         if m.status=3 then
               // The current input and output differences constitute an
                  impossible differential.
15             Store current input and output differences;
```

Constraints for S-box operation Assume S is an arbitrary $m \times l$ bits S-box that $(y_0, y_1, \ldots, y_{l-1}) = S(x_0, x_1, \ldots, x_{m-1})$, the set of all its differential patterns is $DT = \{(\Delta x_0, \ldots, \Delta x_{m-1}, \Delta y_0, \ldots, \Delta y_{l-1}) | Pr[(\Delta x_0, \ldots, \Delta x_{m-1}) \xrightarrow{S} (\Delta y_0, \ldots, \Delta y_{l-1})] > 0\}$. According to Sun *et al.*'s work in [30], we can build linear inequalities system to exactly depict DT, i.e., all possible differentials of S, with the help of the software SAGE [7] and the greed algorithm in [32]. For more details, please refer to [32].

Just similar to MILP models in ARX ciphers, we combine all inequalities for each operation in a certain reduced-round cipher and traverse all input and output differences to judge whether the whole inequalities system has solutions or not under each case. If there is a combination of input and output differences that the MILP model is infeasible, then this is an impossible differential. The search process is as same as Algorithm 1.

[7] Inequality_generator() function in the sage.geometry.polyhedron class of SAGE. The website of SAGE is: http://www.sagemath.org/.

2.3 ZC Approximation Model for ARX Ciphers and Ciphers with S-box

As the counterpart of impossible differential cryptanalysis, zero-correlation linear cryptanalysis is a powerful analysis method as well. The search method is similar to that for impossible differentials except that the inequalities corresponding to each operation is different, which is referred from [15] and [32]. We briefly describe this model for ZC approximations as well in Support material A.

3 Algorithm to Verify the Impossible Differentials and ZC Approximations

In this section, we propose an algorithm to verify impossible differentials and zero-correlation linear approximations searched by our new tools.

As we know, both $\mathcal{U}$-method and UID-method use the miss-in-the-middle technique [5]. They need to firstly construct two characteristic matrices for one round encryption and one round decryption, then calculate the difference state after r_1 rounds from the given input difference and calculate the difference state before r_2 round from the given output difference with probability one, if there is a contradiction bertween some bits of these two states, this is a $r_1 + r_2$ rounds impossible differential.

Inspired by the methods above, we propose an algorithm to find the contradictions of impossible differentials and ZC approximations found by MILP method. Taking the impossible differential as an example, we illustrate our idea, so does the process for ZC approximation. Assume that we already found out a R-round impossible differential $\Delta in \nrightarrow \Delta out$ of a cipher with block size n, which means the corresponding MILP model has no solution. Just like the left part of Figure 1, the target cipher is divided into two parts, Part I and Part II, from a suitable middle state. Generally this suitable state is decided at the output of round $\lceil \frac{R}{2} \rceil$ because of the fast propagation of difference. Accordingly, the inequalities system (including equalities) in MILP model can be divided into three parts — Part I (rounds $1 \sim \lceil \frac{R}{2} \rceil$), Part II (rounds $\lceil \frac{R}{2} \rceil + 1 \sim R$) and Part III (equalities linking the output difference of round $\lceil \frac{R}{2} \rceil$ and input difference of round $\lceil \frac{R}{2} \rceil + 1$) shown in the right part of Figure 1. If inequalities in Part III are all removed from the MILP model, then the remained model must have solutions, which means the two solution sets on the n-bit output of round $\lceil \frac{R}{2} \rceil$ computed from Δin and Δout respectively have no intersection. However, n is usually so large that the two sets are hard to get with limited computing resource, so the contradiction is expected to happen on a small set, such as t bits, where $t << n$. This means when $n - t$ equalities in Part III are removed, the model is still infeasible. Once the position of such t-bit contradiction is found, a model exactly describing rounds $1 \sim \lceil \frac{R}{2} \rceil$ with fixed input difference Δin and a model exactly describing rounds $\lceil \frac{R}{2} \rceil + 1 \sim R$ with fixed output difference Δout can be produced, then two solution sets on this t bits according to the two corresponding models can be calculated. These two sets should be contradictory,

and they never have intersection. Now we can intuitively verify the impossible differential. The whole verification process is summarized in Algorithm 2.

For the search of related-key impossible differentials for a target cipher, the set of inequalities for key schedule and constraints on master key is regarded as the Part IV. In the verification process, Part IV should be put into Part I and Part II simultaneously. The remained process is as same as Algorithm 2.

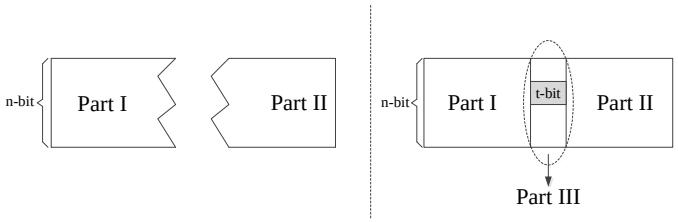

Fig. 1. Brief description to find the contradiction. The left of this figure means a cipher is divided into two parts from a suitable middle state. The right one, which exactly describe the left cipher, indicates the inequalities system are correspondingly divided into three parts. Usually Part III involves equalities.

Algorithm 2: Verification proccess of an impossible differential

Require:
R-round impossible differential $\Delta in \nrightarrow \Delta out$;
Ensure:
Position of t-bit contradiction and two contradictory sets there;
1: Decide the suitable middle state to observe the contradiction;
2: Try to remove as many as equalities in Part III of MILP model but the model is still infeasible, assume t equalities are remained;
3: Produce a model from round 1 to $\lceil \frac{R}{2} \rceil$ with fixed input difference Δin;
4: Produce a model from round $\lceil \frac{R}{2} \rceil + 1$ to R with fixed output difference Δout;
5: Traverse all differences on the t bits in the first model, and put the t-bit differences making the model to have solution into a given list;
6: Handle the second model similarly to step 5, but put the possible t-bit differences into another given list;
7: Judge if the two lists have intersection or not.
8: **return** Position of the t bits contradiction and two lists;

4 Application to HIGHT Block Cipher

4.1 Brief Description of HIGHT

HIGHT, introduced by Hong *et al.* at CHES 2006 [16], is a lightweight block cipher approved by Korea Information Security Agency (KISA) and is adopted as an International Standard by ISO/IEC 18033-3 [17]. Its block size and key size are 64 bits and 128 bits respectively. HIGHT employs the Type-II generalized Feistel network consisting of 32 rounds with four parallel Feistel functions in each round. Whitening keys are applied before the first round and after the last round. The round function is shown in Figure 2, where $(X_7^i|X_6^i,\ldots,|X_0^i)$ and $(SK_{4i+3}|SK_{4i+2}|SK_{4i+1}|SK_{4i})$ indicate the 64 bits input and 32 bits subkey of the i-th round respectively.

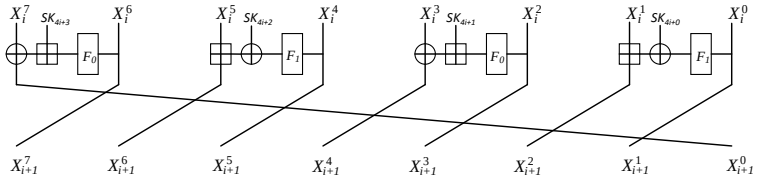

Fig. 2. Round function of HIGHT cipher

Denote exclusive-or, addition modulo 2^{32} and left rotation operations as $\oplus$, $\boxplus$ and $\lll$ respectively. F_0 and F_1, used in the round function, are defined as follows:

$$F_0(x) = (x \lll 1) \oplus (x \lll 2) \oplus (x \lll 7),$$
$$F_1(x) = (x \lll 3) \oplus (x \lll 4) \oplus (x \lll 6).$$

Since the key schedule is not related to the search of impossible differentials and zero-correlation linear approximations, we omit it in this paper. For further details, please refer to [16].

4.2 17-Round Impossible Differentials of HIGHT

For HIGHT block cipher, the longest impossible differential, firstly proposed by Lu in [22], is 16 rounds. Based on the property that the modular addition $\boxplus$ operation definitely preserves the least significant difference in the original positions, he exploited the miss-in-the-middle manner [5] to find two impossible differentials for 16 rounds HIGHT cipher as follows.

$$(e_{j,\sim}, 0^8, 0^8, 0^8, 0^8, 0^8, 0^8, 0^8) \nrightarrow (e_{0,3,5,6,7}, 0^8, 0^8, 0^8, 0^8, 0^8, 0^8, e_7)$$

$$(e_7, e_{0,3,5,6,7}, 0^8, 0^8, 0^8, 0^8, 0^8, 0^8) \nrightarrow (0^8, e_{j,\sim}, 0^8, 0^8, 0^8, 0^8, 0^8, 0^8)$$

Where e_j denotes a byte with zeros in all positions except bit j, $e_{i_1,\ldots,i_j}$ denotes $e_{i_1} \oplus \ldots \oplus e_{i_j}$, $e_{j,\sim}$ denotes a byte that has zeros on bits 0 to $j-1$, 1 on bit j and indeterminate values on bits $(j+1)$ to 7, 0^8 denotes a zero byte.

In this part, we use the form of inequalities system described exactly for modular addition, XOR and bit rotation operations in section 2 to build a MILP model for 17-round HIGHT cipher. Since traversing all input and output differences is impossible due to the time complexity, we only try the cases that the hamming weights of both input and output differences are exactly one, we find four 17-round impossible differentials as follows.

$$(10000000, 0^8, 0^8, 0^8, 0^8, 0^8, 0^8, 0^8) \nrightarrow (0^8, 0^8, 0^8, 0^8, 0^8, 0^8, 0^8, 10000000),$$

$$(0^8, 0^8, 10000000, 0^8, 0^8, 0^8, 0^8, 0^8) \nrightarrow (0^8, 10000000, 0^8, 0^8, 0^8, 0^8, 0^8, 0^8),$$

$$(0^8, 0^8, 0^8, 0^8, 10000000, 0^8, 0^8, 0^8) \nrightarrow (0^8, 0^8, 0^8, 10000000, 0^8, 0^8, 0^8, 0^8),$$

$$(0^8, 0^8, 0^8, 0^8, 0^8, 0^8, 10000000, 0^8) \nrightarrow (0^8, 0^8, 0^8, 0^8, 0^8, 10000000, 0^8, 0^8).$$

These impossible differentials are searched out with Gurobi 6.0.4. on a server using 12 threads Intel(R) Xeon(R) CPU E5-2620(2.00GHz, 47GB RAM, Ubuntu 14.04.3 LTS). Totally it costs 4445 seconds (about 74 minutes). The code can be obtained from: https://github.com/csy1234/NewAutomaticSearchTool/tree/master/HIGHT-IDC/FindTrail.

Taking the first impossible differential as an example, we verify it by using the algorithm in section 3. One contradiction is found on the last output byte of round 9 (input of round 10). The first solution set on this 8-bit contradiction calculated with model for round $1 \sim 9$ includes 255 possible values except 10000000, and the second solution set on this 8-bit contradiction calculated with model for round $10 \sim 17$ only have the value 10000000. This means these two sets have no intersection, further means this 17-round impossible differential is right.

4.3 17-Round ZC Approximations of HIGHT

Until now, for the HIGHT block cipher, the longest ZC approximation is 16 rounds presented by Wen *et al.* in [34], which utilized the mask property of addition that the correlation is not zero if and only if two input masks and output mask have the same high non-zero bit position in [11]. They tried to set the non-zero bits of mask on the highest position of each branch of input and output, and found 128 zero approximations, see Theorem 1 in [34].

In this part we utilize the MILP model proposed in section A to search longer ZC approximations for HIGHT cipher. In this model, the masks of all subkeys are set as free variables. Because of the time complexity as well, we only try the cases that the hamming weights of both input and output masks are exactly one, and we found four 17-round zero-correlation linear approximations as follows.

$$(0^8, 00000001, 0^8, 0^8, 0^8, 0^8, 0^8, 0^8) \nrightarrow (00000001, 0^8, 0^8, 0^8, 0^8, 0^8, 0^8, 0^8),$$

$$(0^8, 0^8, 0^8, 00000001, 0^8, 0^8, 0^8, 0^8) \nrightarrow (0^8, 0^8, 00000001, 0^8, 0^8, 0^8, 0^8, 0^8),$$
$$(0^8, 0^8, 0^8, 0^8, 0^8, 00000001, 0^8, 0^8) \nrightarrow (0^8, 0^8, 0^8, 0^8, 00000001, 0^8, 0^8, 0^8),$$
$$(0^8, 0^8, 0^8, 0^8, 0^8, 0^8, 0^8, 00000001) \nrightarrow (0^8, 0^8, 0^8, 0^8, 0^8, 0^8, 00000001, 0^8).$$

These ZC approximations are found with Gurobi 6.0.4. on a server using 12 threads Intel(R) Xeon(R) CPU E5-2620(2.00GHz, 47GB RAM, Ubuntu 14.04.3 LTS). Totally it costs 4786 seconds(about 80 minutes). The code can be obtained from website: https://github.com/csy1234/NewAutomaticSearchTool/tree/master/HIGHT-ZC/FindTrail.

Taking the second ZC approximation as example, one contradiction is found on the first output byte of round 9 (input of round 10). The first solution set on this 8-bit contradiction calculated from the fixed input mask involves 255 values except 00000001, and the set solution set on this 8-bit contradiction calculated from the fixed output mask only has one 00000001. This means such approximation is really a zero-correlation linear approximation.

5 Application to LBlock Cipher

5.1 Brief Description of LBlock

LBlock, designed by Wu and Zhang at ACNS in [39], is a lightweight block cipher. On account of its excellent hardware performance, software performance and security, it is widely focused on by the cryptanalysts in the field of symmetric cryptography. Its block size and key size are 64 bits and 80 bits respectively. LBlock cipher adopts a 32-round modified Feistel network which adds an extra left rotation operation on one branch of general Feistel network. The round function is shown in Figure 3, where (X_1^i, X_0^i) and sk_i denote 64 bits input and 32 bits subkey of the i-th round respectively.

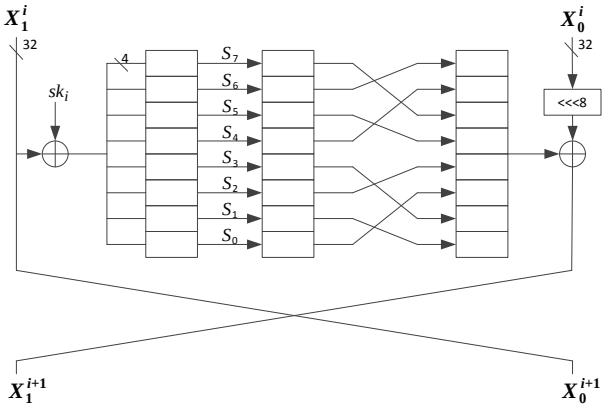

Fig. 3. Round function of LBlock cipher

In the round function, there are a xor operation with subkey, a nonlinear layer and a simple permutation that the second component involves 8 parallel different S-boxes $S_0, S_1, S_2, S_3, S_4, S_5, S_6, S_7$ and the last component only changes the byte order of its input. It is worth noting that an 8-bit left rotation operation happens on the right branch in Figure 3.

Key Schedule The master key of LBlock cipher is 80 bits, denoted by $K = k_{79}, k_{78}, \ldots, k_0$. All subkeys $sk_i, i = 0, 1, \ldots, 31$ are produced by utilizing a 80-bit register. The process is illustrated in Algorithm 3.

Algorithm 3: Key schedule of LBlock cipher

$sk_0 = K_{79\sim 48}$;
for $1 \leq r \leq 31$ **do**
  $k_{79\sim 0} \leftarrow k_{79\sim 0} \lll 29$;
  $k_{79\sim 76} \leftarrow S_9(k_{79\sim 76})$;
  $k_{75\sim 72} \leftarrow S_8(k_{75\sim 72})$;
  $k_{50\sim 47} \leftarrow k_{50\sim 47} \oplus [i]_2$;
  $sk_r \leftarrow k_{79\sim 48}$

In Algorithm 3, $k_{a\sim b}$ denoted all key bits from k_a to k_b, S_8 and S_9 are two different 4×4 S-boxes. For more details about LBlock, please refer to [39].

5.2 16-Round Related-Key Impossible Differentials of LBlock

Differential cryptanalysis and impossible differential cryptanalysis are both implemented under the single-key setting, i.e., all plaintexts are encrypted with one master key. In [1] and [18], related-key differential and related-key impossible differential cryptanalysis are proposed respectively, which exploit the relation of two master keys to recover the secret keys.

For LBlock cipher, Minier and Naya-Plasencia found a 15-round related-key impossible differential and attacked 22-round LBlock with it in [36]. In [35], Wen *et al.* designed a specialized algorithm to search longer related-key impossible differentials with some observations on key schedule and structure of the cipher. They totally found two 16-round related-key impossible differentials. However, Wen *et al.*'s two impossible differentials are right only under part of master key pairs which satisfy one of the given two key differences.

In our work, we use the method in Section 2 to build the MILP model for LBlock cipher including the key schedule and search the related-key impossible differentials. Considering the key schedule, LBlock is a bit-level cipher. We only search the cases that the difference of master key has only one nonzero bit (80 cases) and the input and output differences both have no more than one nonzero bit ($65 \times 65 = 4225$ cases), so in total we search 338000 cases. In the end, we

search out six 16-round related-key impossible differentials, whose forms are as follows:

$$0 \xrightarrow{16r,\Delta K} 0,$$

where the input and output differences are both zero and the difference of master key ΔK has only one nonzero bit difference among $k_0, k_1, k_2, k_6, k_{10}, k_{11}$ (six cases). Unlike Wen *et al.*'s work, as long as the master key pair satisfies one of the six differences above, the related-key impossible differential is right in our work.

These six related-key impossible differential are found with Gurobi 6.0.4. on a server using 12 threads Intel(R) Xeon(R) CPU E5-2620(2.00GHz, 47GB RAM, Ubuntu 14.04.3 LTS). It costs several hours.

Taking the related-key impossible differential with k_0 as example, one contradiction is found on the sixth most significant output nibble of round 8 (input of round 9). The first solution set on this 4-bit contradiction calculated from the fixed input difference involves only one values 0000, and the set solution set on this 4-bit contradiction calculated from the fixed output difference only has 11 values: 1000, 0100, 1100, 1010, 0110, 1110, 0001, 0101, 0011, 1011, 1111. This means such approximation is really a related-key impossible differential.

6 Conclusion

In this paper, we propose an automatic search tool for impossible differentials and ZC approximations based on MILP method. In this tool, the differential and linear properties of non-linear components are taken into consideration, so we can not only find the previous impossible differentials and ZC approximations, but also may find longer ones for a target cipher. As applications, we apply this tool on HIGHT and LBlock ciphers. As a result, we find 4 impossible differentials and 4 ZC approximations for 17-round HIGHT, which are the longest ones for HIGHT cipher until now, and find six 16-round related-key impossible differentials for LBlock. There is a problem as well. Since 8×8 S-box cannot be exactly described with a set of inequalities in MILP model, which is a limitation of our automatic search tool, in the further we will go on researching how to apply this tool on ciphers with large size of S-box.

References

1. Biham, E.: New types of cryptanalytic attacks using related keys. Journal of Cryptology, 7(4): 229-246. (1991)
2. Biham, E., Biryukov, A., Shamir, A.: Cryptanalysis of Skipjack reduced to 31 rounds using impossible differentials. In International Conference on the Theory and Applications of Cryptographic Techniques (pp. 12-23). Springer Berlin Heidelberg. (1999, May)
3. Biham, E., Shamir, A.: Differential cryptanalysis of the data encryption standard. Springer Science Business Media. (2012)

4. Biham, E.: On Matsui's linear cryptanalysis. In Workshop on the Theory and Application of of Cryptographic Techniques (pp. 341-355). Springer Berlin Heidelberg. (1994, May)
5. Biham, E., Biryukov, A., Shamir, A.: Miss in the Middle Attacks on IDEA and Khufu. In International Workshop on Fast Software Encryption (pp. 124-138). Springer Berlin Heidelberg. (1999, March)
6. Blondeau, C.: Impossible differential attack on 13-round Camellia-192. Information Processing Letters, 115(9), pp. 660-666. (2015)
7. Bogdanov, A., Rijmen, V.: Linear hulls with correlation zero and linear cryptanalysis of block ciphers. Designs, codes and cryptography, 70(3), pp. 369-383. (2014)
8. Bogdanov, A., Wang, M.: Zero correlation linear cryptanalysis with reduced data complexity. In Fast Software Encryption (pp. 29-48). Springer Berlin Heidelberg. (2012)
9. Bogdanov, A., Leander, G., Nyberg, K., Wang, M.: Integral and multidimensional linear distinguishers with correlation zero. In International Conference on the Theory and Application of Cryptology and Information Security (pp. 244-261). Springer Berlin Heidelberg. (2012, December)
10. Bogdanov, A., Geng, H., Wang, M., Wen, L., Collard, B.: Zero-correlation linear cryptanalysis with FFT and improved attacks on ISO standards Camellia and CLEFIA. In International Conference on Selected Areas in Cryptography (pp. 306-323). Springer Berlin Heidelberg. (2013, August)
11. Bogdanov, A., Wang, M.: Zero correlation linear cryptanalysis with reduced data complexity. In Fast Software Encryption (pp. 29-48). Springer Berlin Heidelberg. (2012)
12. Boura, C., Naya-Plasencia, M., Suder, V.: Scrutinizing and improving impossible differential attacks: applications to CLEFIA, Camellia, LBlock and Simon. In International Conference on the Theory and Application of Cryptology and Information Security (pp. 179-199). Springer Berlin Heidelberg. (2014, December)
13. Chen, J., Wang, M., Preneel, B.: Impossible Differential Cryptanalysis of Lightweight Block Ciphers TEA, XTEA and HIGHT. IACR Eprint Archive Report 2011/616 (2011)
14. Derbez, P., Fouque, P.-A.: Automatic Search of Meet-in-the-Middle and Impossible Differential Attacks. CRYPTO 2016, Part II, LNCS 9815, pp. 157184, 2016. DOI: 10.1007/978-3-662-53008-5_6
15. Fu, K., Wang, M., Guo, Y., Sun, S., Hu, L.: MILP-Based Automatic Search Algorithms for Diff erential and Linear Trails for Speck. representations, 21, 27. (2016)
16. Hong, D., Sung, J., Hong, S., Lim, J., Lee, S., Koo, B. S., ... Kim, H.: HIGHT: A new block cipher suitable for low-resource device. In International Workshop on Cryptographic Hardware and Embedded Systems (pp. 46-59). Springer Berlin Heidelberg. (2006, October)
17. ISO/IEC 18033-3, Information technology-Security techniques-Encryption algorithms-Part 3: Block ciphers (2010)
18. Jakimoski, G., Desmedt, Y.: Related-key differential cryptanal- ysis of 192-bit key AES variants. In Proc. the 10th Annual International Workshop on Selected Areas in Cryptography, pp.208-221. (2003, August)
19. Kim, J., Hong, S., Lim, J.: Impossible differential cryptanalysis using matrix method. Discrete Mathematics, 310(5), 988-1002. (2010)
20. Knudsen, L.: DEAL-a 128-bit block cipher. complexity, 258(2), 216. (1998)
21. Lipmaa, H., Moriai, S.: Efficient algorithms for computing differential properties of addition. In International Workshop on Fast Software Encryption (pp. 336-350). Springer Berlin Heidelberg. (2001, April)

22. Lu, J.: Cryptanalysis of reduced versions of the HIGHT block cipher from CHES 2006. In International Conference on Information Security and Cryptology (pp. 11-26). Springer Berlin Heidelberg. (2007, November)
23. Luo, Y., Lai, X., Wu, Z., Gong, G.: A unified method for finding impossible differentials of block cipher structures. Information Sciences, 263, pp. 211-220. (2014)
24. Mala, H., Dakhilalian, M., Rijmen, V., Modarres-Hashemi, M.: Improved impossible differential cryptanalysis of 7-round AES-128. In International Conference on Cryptology in India (pp. 282-291). Springer Berlin Heidelberg. (2010, December)
25. Matsui, M.: Linear cryptanalysis method for DES cipher. In Workshop on the Theory and Application of of Cryptographic Techniques (pp. 386-397). Springer Berlin Heidelberg. (1993, May)
26. Mouha, N., Wang, Q., Gu, D., Preneel, B.: Differential and linear cryptanalysis using mixed-integer linear programming. In International Conference on Information Security and Cryptology (pp. 57-76). Springer Berlin Heidelberg. (2011, November)
27. Nyberg, K., Wallén, J.: Improved linear distinguishers for SNOW 2.0. In International Workshop on Fast Software Encryption (pp. 144-162). Springer Berlin Heidelberg. (2006, March)
28. Özen, O., Varici, K., Tezcan, C., Kocair, Ç.: Lightweight Block Ciphers Revisited: Cryptanalysis of Reduced Round PRESENT and HIGHT. ACISP 2009. LNCS, vol. 5594, pp. 90107. Springer. (2009)
29. Sun, B., Liu, Z., Rijmen, V., Li, R., Cheng, L., Wang, Q., ... Li, C.: Links among impossible differential, integral and zero correlation linear cryptanalysis. In Annual Cryptology Conference (pp. 95-115). Springer Berlin Heidelberg. (2015, August)
30. Sun, S., Hu, L., Wang, P., Qiao, K., Ma, X., Song, L.: Automatic security evaluation and (related-key) differential characteristic search: application to SIMON, PRESENT, LBlock, DES (L) and other bit-oriented block ciphers. In International Conference on the Theory and Application of Cryptology and Information Security (pp. 158-178). Springer Berlin Heidelberg. (2014, December)
31. Sun, S., Hu, L., Song, L., Xie, Y., Wang, P.: Automatic security evaluation of block ciphers with S-bP structures against related-key differential attacks. In International Conference on Information Security and Cryptology (pp. 39-51). Springer International Publishing. (2013, November)
32. Sun, S., Hu, L., Wang, M., Wang, P., Qiao, K., Ma, X., ... Fu, K.: 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)
33. Wallén, J.: Linear approximations of addition modulo 2^n. In International Workshop on Fast Software Encryption (pp. 261-273). Springer Berlin Heidelberg. (2003, February)
34. Wen, L., Wang, M., Bogdanov, A., Chen, H.: Multidimensional zero-correlation attacks on lightweight block cipher HIGHT: Improved cryptanalysis of an ISO standard. Information Processing Letters, 114(6), 322-330. (2014)
35. Wen, L., Wang, M., and Zhao, J.: Related-Key Impossible Differential Attack on Reduced-Round LBlock. J. Comput. Sci. Technol., 29(1):165176. (2014)
36. Minier, M., Naya-Plasencia, M.: A related key impossible differ- ential attack against 22 rounds of the lightweight block cipher LBlock. Information Processing Letters, 112(16): pp. 624- 629. (2012)
37. Wu, S., Wang, M.: Security Evaluation against Differential Cryptanalysis for Block Cipher Structures. IACR Cryptology ePrint Archive, 2011, 551. (2011)

38. Wu, S., Wang, M.: Automatic search of truncated impossible differentials for word-oriented block ciphers. In International Conference on Cryptology in India (pp. 283-302). Springer Berlin Heidelberg. (2012, December)
39. Wu, W., Zhang, L.: LBlock: A lightweight block cipher. In Proc. the 9th International Conference on Applied Cryptography and Network Security, pp.327-344. (2011)

A Appendix

A.1 Zero-Correlation Linear Model for ARX Cipher

In order to search ZC approximations in ARX ciphers, it is necessary to consider about the linear approximations of basic operations such as XOR, branching, bit rotation and modular addition operations. Before studying the construction of MILP model for search of ZC approximations, we introduce the linear approximations over XOR and branching operations proposed by Biham in [4] as follows, where "$\cdot$" means the scalar product of binary vectors.

Lemma 1 (XOR operation [4]). *Let $h(x_1, x_2) = x_1 \oplus x_2$, α_1, α_2 are the input masks of x_1 and x_2 respectively, β is the output mask, then the correlation $C(\beta \cdot h(x_1, x_2), \alpha_1 \cdot x_1 \oplus \alpha_2 \cdot x_2) \neq 0$ if and only if $\beta = \alpha_1 = \alpha_2$.*

Lemma 2 (Branching operation [4]). *Let $h(x) = (x, x)$, α is the input mask, β_1, β_2 are the output masks of $h(x)$, then the correlation $C((\beta_1, \beta_2) \cdot h(x), \alpha \cdot x) \neq 0$ if and only if $\alpha = \beta_1 \oplus \beta_2$.*

Following the Lemma 1 and 2, we start to construct the MILP model for search of ZC approximations in ARX ciphers.

Constraints for Branching, XOR and Bit Rotation Assumed that the input mask of braching operation is α, the output masks are β_1 and β_2. According to Lemma 2, $\alpha = \beta_1 \oplus \beta_2$, so similar to (2) in section 2.1, we have the following equality to exactly describe its each bit operation.

$$\alpha[i] + \beta_1[i] + \beta_2[i] = 2d_{\oplus} \tag{5}$$

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

In the light of Lemma 1, some linear equations between input masks and output mask can perfectly describe the linear approximation of XOR operation. Besides, the bit rotation operation is a simple permutation that we can list some equations for the related bits.

Constraints for Modular Addition In [33, 27], a method to calculate the correlation of modular addition is given as follows.

Theorem 2 ([33, 27]). *For the linear approximation of addition modulo 2^n, let the input masks and output mask be $\alpha_1 = (\alpha_1[n-1], \ldots, \alpha_1[0])$, $\alpha_2 = (\alpha_2[n-1], \ldots, \alpha_2[0])$ and $\beta = (\beta[n-1], \ldots, \beta[0])$ respectively, where $\alpha_1, \alpha_2, \beta \in \mathcal{F}_2^n$, and let the vector $u = (u[n-1], \ldots, u[0])$ satisfy $u[i] = 4\beta[i] + 2\alpha_1 + \alpha_2, 0 \leq u[i] < 8, 0 \leq i < n$. Then the correlation can be computed as follows:*

$$cor_{\boxplus}(\beta, \alpha_1, \alpha_2) = LA_{u[n-1]}A_{u[n-2]} \ldots A_{u[0]}C, \tag{6}$$

where $A_r, 0 \leq r < 7$, is 2×2 matrice,

$$A_0 = \frac{1}{2}\begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}, A_1 = A_2 = -A_4 = \frac{1}{2}\begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix},$$

$$A_7 = \frac{1}{2}\begin{bmatrix} 0 & 2 \\ 1 & 0 \end{bmatrix} - A_3 = A_5 = -A_6 = \frac{1}{2}\begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix},$$

L is a row vector $L = (1, 0)$, and C is a column vector $C = (1, 1)^T$.

In order to quickly calculate the correlation shown in Theorem 2, Nyberg and Wellén utilized the automaton to calculate (6) by multiplication from left to right [27]. They let $e_0 = L = (1, 0)$ and $e_1 = (0, 1)$, then the state transitions for addition modulo 2^n is as follows:

$$\varepsilon_n = e_0 \xrightarrow{u[n-1]} \varepsilon_{n-1} \xrightarrow{u[n-2]} \varepsilon_{n-2} \rightarrow \ldots \rightarrow \varepsilon_1 \xrightarrow{u[0]} \varepsilon_0.$$

Where $\varepsilon_j \in \{e_0, e_1\}, 0 \leq j < n$. For more details, please refer to [27].

Based on the work above, Fu *et al.* in [15] set a $0-1$ variable $s_i = 0$ if $\varepsilon_i = e_0$, otherwise $s_i = 1$, then utilized $(s_{i+1}, \beta[i], \alpha_1[i], \alpha_2[i], s_i)$ to describe the state transition from ε_{i+1} to ε_i, namely $e_{s_{i+1}}A_{u[i]} = e_{s_i}$. They found that there are 10 possible transitions for the vector $(s_{i+1}, \beta[i], \alpha_1[i], \alpha_2[i], s_i)$, and listed eight linear inequalities exactly satisfying these 10 possible transitions with the help of SAGE and the greedy algorithm in [32], which are shown as follows:

$$\begin{aligned}
&s_{i+1} - \beta[i] - \alpha_1[i] + \alpha_2[i] + s_i \geq 0, \quad && s_{i+1} + \beta[i] + \alpha_1[i] - \alpha_2[i] - s_i \geq 0,\\
&s_{i+1} + \beta[i] - \alpha_1[i] - \alpha_2[i] + s_i \geq 0, \quad && s_{i+1} - \beta[i] + \alpha_1[i] - \alpha_2[i] + s_i \geq 0,\\
&s_{i+1} + \beta[i] - \alpha_1[i] + \alpha_2[i] - s_i \geq 0, \quad && s_{i+1} - \beta[i] + \alpha_1[i] + \alpha_2[i] - s_i \geq 0,\\
&-s_{i+1} + \beta[i] + \alpha_1[i] + \alpha_2[i] + s_i \geq 0, \quad && s_{i+1} + \beta[i] + \alpha_1[i] + \alpha_2[i] + s_i \geq 0.
\end{aligned}$$

Note that there is an additional constraint $\varepsilon_n = e_0$, hence, the constraints include $8 \times n + 1$ linear inequalities for linear approximation of addition modulo 2^n.

Until now, every operation in a certain reduced-round ARX cipher is described with inequalities. The corresponding MILP model for search of ZC approximations is built by combining the whole inequalities system of all operations, and it is as same as the building process for search of impossible differentials. If we already have the MILP model file "model.lp" for a target cipher, the general search process is similar to the Algorithm 1 when we use Gurobi as the optimization except that we use mask in ZC approximations instead of difference in impossible differentials.

A.2 Zero-Correlation Linear Model for Ciphers with S-box

Since the linear operations used in ciphers with S-box are as same as those in ARX ciphers, for the sake of simplicity, we omit them in this subsection.

Constraints for S-box operation Assume S is an arbitrary $m \times l$ bits S-box that $(y_0, y_1, \ldots, y_{l-1}) = S(x_0, x_1, \ldots, x_{m-1})$, and α, β are input and output masks respectively, then the set of all its meaningful linear approximations is $LT = \{(\alpha, \beta) | Pr[\alpha \xrightarrow{S} \beta] \neq \frac{1}{2}\}$. Similar to the construction of constraints for S-box in impossible differential cryptanalysis in section 2.2, we can build linear inequalities system to exactly depict the set of LT, with the help of SAGE and Greedy algorithm in [32].

Next, we still combine all inequalities for each operation in a certain reduced-round cipher with S-box. Then the search process is as same as that for ARX ciphers.