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 may 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 proof 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 the proposed automatic tool on HIGHT and PRESENT ciphers. As a result, we find total 4 impossible differentials and 4 zero-correlation linear approximations for 17-round HIGHT which are the longest ones until now. In addition, we find 5050 impossible differentials for 6-round PRESENT cipher, which extend two more rounds than those searched by previous widely applicable automatic search tools.

Keywords: Automatic search tool, impossible differential, zero-correlation linear, HIGHT, PRESENT

1 Introduction

Impossible differential cryptanalysis (IDC), introduced by Biham *et al.* and Knudsen to attack Skipjack in [1] and DEAL [17] respectively, unlike the differential cryptanalysis [2] which goal is to find a differential characteristic with high probability, is to find the longest impossible differential, i.e., to find the longest differential with probability 0. It is a powerful cryptanalysis tool. Since its establishment, lots of block ciphers are encountered with it such as AES [21], LBlock [12], Camellia [5, 12] and so on. As the counterpart of impossible differential cryptanalysis, Zero-correlation linear cryptanalysis, a variant of linear cryptanalysis [22], is proposed by Bogdanov *et al.* in [7] and improved in [8–11]. Similar to the idea of impossible differential, its aim is to find a linear approximation with probability exactly 1/2. In [25], Sun *et al.* proposed that in some cases, a zero-correlation linear approximations is 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 [16], UID-method [20] and the extended tool of them generalized by Wu *et al.* [33]. Moreover, it has been proved that the last method of [33] can find all impossible differentials of a structure in [25], i.e., can find all impossible differentials of a block cipher which are independent of the non-linear components such as S-box. However, the differential property in S-box is wasted in above-mentioned methods. Especially 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 will 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 [26, 13], we hope to search the impossible differentials 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 [23] and [32] respectively, later improved in [27, 26, 13, 28]. According to its applications on the search of differential 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 fast run out the feasible and optimized solutions, we can search the optimal characteristic for the target cipher with suitable implement time. When traversing all input and output differences (masks in linear cryptanalysis), can MILP method be used to the search of impossible differentials with suitable time? This is the motivation for us to do this work.

1.1 Contributions

Propose an automatic search tool for impossible differentials in both ARX ciphers and ciphers with S-box. Impossible differential cryptanalysis is one of efficient cryptanalysis methods in the field of symmetric ciphers. Up to now, several automatic search approaches have been proposed for it such as $\mathcal{U}$-method, UID-method and the improved method in [33], which are all independent of the non-linear components such as S-box. However, in fact the differential properties of non-linear operations can also contribute to the search of longer impossible differentials. In this paper, in order to solve this defect in above-mentioned methods, we propose a new automatic search tool for impossible differentials in ARX ciphers and ciphers with S-box in [13] and [26]. With this new tool, all operations are considered including the differential properties of non-linear operations, so we may find longer impossible differentials comparing with the previous methods. In addition, by traversing a special subset of input and output differences depending on the actual cipher, this method can proof whether there is an impossible differential 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 a longer impossible differential.

Propose an automatic search tool for zero-correlation linear approximations in both ARX ciphers and ciphers with S-box. Zero-correlation linear cryptanalysis is another useful cryptanalysis method. Very similar to the automatic search algorithms for impossible differentials, we can also use $\mathcal{U}$-method, UID-method and the improved method in [33] to search linear approximations with probability exactly 1/2. However, the same problem handled for non-linear operations is still existed. Based on the work proposed by Sun *et al.* in [26] for cipher Ciphers with S-boxs and Fu *et al.* in [13] for ARX ciphers, we also present an automatic tool based for search of zero-correlation linear approximations. This method is the first widely applicable automatic search tool with considering the linear properties of non-linear components. With this tool, it is more possible to find a longer zero-correlation linear approximation. What's more, it can be used to proof that whether there is a zero-correlation linear approximation or not for a special input and output pattern of a given cipher.

Application to HIGHT cipher. HIGHT cipher, introduced by Hong *et al.* at CHES 2006 [14], is an ISO standard lightweight block cipher. Its block size and key size are 64 bits and 128 bits respectively, and it has total of 32 rounds. Until now, the longest impossible differential and zero-correlation linear approximation are both 16 rounds, which are introduced by Lu in [19] and Wen *et al.* in [31] respectively. In our work, we use the proposed automatic tool to search all cases of 17-round impossible differentials (zero-correlation linear approximations) that both hamming weights of input and output differences (masks) are one. As a result, we find total 4 impossible differentials and 4 zero-correlation linear approximations for 17-round HIGHT, which are the longest ones until now.

Application to PRESENT cipher. PRESENT cipher, proposed by Bogdanov *et al.* at CHES 2007 [6], is a lightweight block cipher designed for hardware constrained environments such as RFID tags and sensor networks. It adopts

SP-network and consists of 31 rounds. Its block size is 64 bits and two key sizes of 80 and 128 bits are supported. Until now, the longest impossible differential is proposed by Tezcan in [29]. He found one 6-round impossible differential by using some special differential property of S-box. But with existing widely applicable automatic search tools such as $\mathcal{U}$-method, UID-method and their improved method in [33], it is only possible to obtain 4-round impossible differentials for PRESENT ciphers. In this paper, we use this automatic search tool to find out 5050 impossible differentials for 6 rounds of PRESENT cipher.

Application to the existence proof. With this automatic search tool, we can proof the existence of impossible differentials and zero-correlation linear approximations for certain rounds of block ciphers in a special patterns of input and output differences (masks). Usually, the less hamming weights input and output differences (mask) have, the more advantage one has to find a better impossible differential (zero-correlation linear approximation), so we can apply this tool on existence proof of impossible differentials and zero-correlation linear approximations under a subset cases, i.e., specially patterns. As applications, we proof that the longest impossible differentials for LBlock, TWINE and Piccolo ciphers are really 14, 14 and 7 rounds respectively when we only consider the patterns that the 8 nibbles of input difference which will enter the first round function are all zero and within the other 8 nibbles only one is non-zero difference, meanwhile, the cases on output difference are similar to that on input. By setting the hamming weights of both input and output differences (masks) are one, we find the longest impossible differentials (zero-correlation linear approximations) are 15, 15 rounds for TEA and XTEA and ciphers respectively, and find the longest zero-correlation linear approximations are all 6 rounds for SPECK-32 and SPECK-48 ciphers.

1.2 Outline

This paper is organized as follows. In section 2 and 3, we propose automatic tools for search of impossible differentials and zero-correlation linear approximations in both ARX ciphers and ciphers with S-box. As applications, we use this tool to search longer impossible differentials and zero-correlation linear approximations for HIGHT in section 4 and impossible differentials for PRESENT ciphers in section 5. Then we utilize this tool on the proof existence in section 6. Finally, section 7 concludes the paper.

2 Automatic Tool for Search of Impossible Differentials

Impossible differential cryptanalysis, unlike the differential cryptanalysis which goal is to find a differential characteristic with high probability, is to find the longest differential with probability 0. Until now, there are three widely applicable methods, $\mathcal{U}$-method, UID-method and the improved one proposed by

Wu *et al.*, to search various impossible differential trails of block cipher structures. However, all of them are not to consider the differential property of S-box. Further more, for ARX block ciphers there is no general algorithms to search impossible differentials.

In this section, we will propose an automatic tool for search of impossible differentials in both ARX ciphers and ciphers with S-box by utilizing the mixed integer linear programming. Like the idea of MILP models for differential cryptanalysis in previous work, for the search of impossible differentials, we firstly utilize linear inequalities to exactly describe every component in the cipher as well. But we are indifferent to the objective function, 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 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, and iterating them over multiple rounds. For each component, 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 [26].

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

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 is $0-1$ 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 [18], Lipmaa and Moriai proposed a method to verify whether a given differential characteristic is possible or not. For sake of simplicity Fu *et al.* summarizes this method into a theorem in [13] as follows:

Theorem 1 (see [18, 13]). *The differential* $(\alpha, \beta \rightarrow \gamma)$ *is possible 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, Fu *et al.* utilized five inequalities in [13] to satisfy it as follows:

$$\begin{array}{r} \alpha[0] + \beta[0] + \gamma[0] \leq 2 \\ \alpha[0] + \beta[0] + \gamma[0] - 2d_{\oplus} \geq 0 \\ d_{\oplus} \geq \alpha[0], d_{\oplus} \geq \beta[0], d_{\oplus} \geq \gamma[0] \end{array} \tag{3}$$

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 [13], Fu *et al.* used 13 inequalities to exactly describe the 56 possible patterns for each $i \in [1, n-1]$ as follows.

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

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

Up to now, we exactly describe every operation in ARX cipher with set of inequalities, in which the variables are usually the input and output differences of corresponding operations. Actually, most variable relates to at least two operations except the input and output differences of the cipher, so if we combine all inequalities for every operation in the target ARX cipher, the inequalities system can perfectly depict the whole cipher, its each solution is a differential characteristic. When we fixed the input and output differences, if the inequalities system is infeasible, it means the probability of current differential is 0, i.e., this is an impossible differential. By traversing a special subset of input and output differences in the MILP model, we can 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. For example, we generally set the non-zero bits on the branch which xores with the output of the first round function in Feistel construction. Without loss of generality, we denote the subset as $(\Delta \rightarrow \Gamma)$, where Δ and Γ are the sets on input and output differences respectively. In Algorithm 1, we explain how to implement the search of impossible differentials with Gurobi Optimization, when we already have MILP model file "model.lp", which is produced as same as that for differential characteristics in [28, 13].

Algorithm 1: General search process for impossible differentials

```
// Assume the block size is $n$.
def mycallback(model,where):
    if where == GRB.Callback.MIP:
        best = model.cbGet(GRB.Callback.MIP_OBJBST)
    if best $\geq$ 0:
        m.terminate()
// mycallback function is to terminate the optimization if a
   solution is already appeared.
for All input differences $\Delta x_i \in \Delta$ do
    for All output differences $\Delta y_j \in \Gamma$ do
        if $i = 0$ and $j = 0$ then
            Add all constraints on the fixed input and output differences into
              "model.lp";
        else
            Change all constraints on the fixed input and output differences into
              "model.lp";
        m=read('model.lp');
        m.optimize(mycallback);
        if m.status=3 then
            // The current input and output differences constitute an
               impossible differential.
            Store current input and output differences;
```

2.2 Impossible Differential Model for Ciphers with S-box

Comparing with ARX ciphers, traditional 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 it has been exactly described in section 2.1.

Constraints for S-box operation Assume S is an arbitrary $m \times l$ bits S-box that $(y_0, y_1, \ldots, y_l) = S(x_0, x_1, \ldots, x_m)$, the set of all its differential patterns is $DT = \{(\Delta x_0, \ldots, \Delta x_m, \Delta y_0, \ldots, \Delta y_l) | Pr[(\Delta x_0, \ldots, \Delta x_m) \xrightarrow{S} (\Delta y_0, \ldots, \Delta y_l)] > 0\}$. According to Sun *et al.*'s work in [26], we can build linear inequalities system to exactly depict the set of DT, i.e., all possible differentials of S, with the help of the software SAGE [5] and the greed algorithm in [28]. For more details, please refer to [28]. But note that the large S-boxes such as size of 8×8 cannot be dealed with now.

Just similar to MILP model 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 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.

3 Automatic Tool for Search of Zero-Correlation Linear Approximations

Zero-correlation linear cryptanalysis, introduced by Bogdanov and Rijmen in [7], is an important tool to evaluate the security of block ciphers. Unlike the linear cryptanalysis, whose aim is to find a linear approximation with high bias, zero-correlation cryptanalysis is expected to find the longest linear approximation holding with probability exactly 1/2. For automatic search of zero-correlation linear approximations, $\mathcal{U}$-method, UID-method and the improved method in [33] can be utilized as well as for the search of impossible differential. Similarly, all these methods overlook the linear properties of non-linear operations such as S-boxes. As far as we know there are also no widely applicable automatic method for ARX ciphers.

In this section, we will use the MILP method to solve the problem above and accurately search zero-correlation linear approximations for both ARX ciphers and ciphers with S-box.

3.1 Zero-Correlation Linear Model for ARX Cipher

In order to search zero-correlation linear approximation 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 zero-correlation linear approximation, we introduce the linear approximations over XOR and branching operations proposed by Biham in [3] as follows, where "·" means the scalar product of binary vectors.

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

Lemma 1 (XOR operation [3]). *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 [3]). *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 zero-correlation linear 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 inequalities 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 perfect 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 [30, 24], a method to calculate the correlation of modular addition is given as follows.

Theorem 2 ([30, 24]). *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 [24]. 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 [24].

Based on the work above, Fu *et al.* in [13] 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 list eight linear inequalities exactly satisfying these 10 possible transitions with the help of SAGE and the greedy algorithm in [28], 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, we can combine all inequalities for every operation in a certain reduced-round ARX cipher. When we fixed the input and output masks in the constraints, if the MILP model is infeasible, this is a zero-correlation linear approximation. By traversing a special subset of input and output masks, we can find the existed zero-correlation linear approximations or proof there is no such approximation with probability exactly 1/2 in this subset. Without loss of generality, we denote the subset as $(\Lambda \rightarrow \Omega)$, where Λ and Ω are the sets on input and output masks respectively. If we already have the MILP model file "model.lp" for a target cipher, the general search process is shown in Algorithm 2, when we use Gurobi as the optimization.

3.2 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 $(y_0, y_1, \ldots, y_l) = S(x_0, x_1, \ldots, x_m)$, 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, i.e., the all possible linear approximations of S, with the help of SAGE and Greedy algorithm in [28].

Algorithm 2: General search process for zero-correlation linear approximations

```
// Assume the block size is $n$.
def mycallback(model,where):
    if where == GRB.Callback.MIP:
      best = model.cbGet(GRB.Callback.MIP_OBJBST)
    if best ≥ 0:
      m.terminate()
for All possible input masks $\alpha_i \in \Lambda$ do
    for All possible output masks $\beta_j \in \Omega$ do
        if $i = 0$ and $j = 0$ then
            Add all constraints on the fixed input and output masks into
              "model.lp";
        else
            Change all constraints on the fixed input and output masks into
              "model.lp";
        m=read('model.lp');
        m.optimize(mycallback);
        if m.status=3 then
            // The current input and output masks constitute an
               zero-correlation linear approximation.
            Store current input and output masks;
```

Next, we still combine all inequalities for each operation in a certain reduced-round traditional cipher and traverse all input and output masks to find whether the inequalities system has solutions or not under each case. If there is a case that the MILP model is infeasible, then this is an zero-correlation linear approximation. This procedure is depicted in Algorithm 2 as well.

4 Application to HIGHT Block Cipher

4.1 Brief Description of HIGHT

HIGHT, introduced by Hong *et al.* at CHES 2006 [14], is a lightweight block cipher approved by Korea Information Security Agency (KISA) and is adopted as an International Standard by ISO/IEC 18033-3 [15]. Its block size and key size are 64 bits and 128 bits respectively. HIGHT employees 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 1, 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.

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

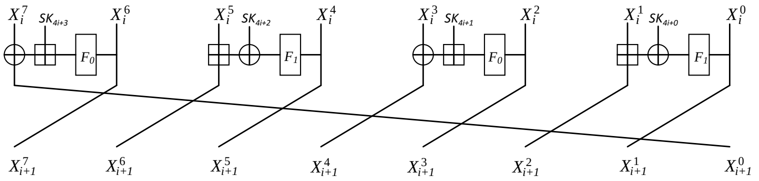

Fig. 1. Round function of HIGHT cipher

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 [14].

4.2 17-Round Impossible Differentials of HIGHT

Since in ARX ciphers none of widely applicable search algorithms have been proposed until now. Up to now, for the HIGHT block cipher, the longest impossible differential, proposed by Lu in [19], is 16 rounds. Based on the property that the modular addition $\boxplus$ operation definitely preserves the least significant differences in the original positions, he exploited the miss-in-the-middle manner [4] 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 but 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 in bits 0 to $j-1$, a one in bit j and indeterminate values in bits $(j+1)$ to 7, 0^8 denotes a byte with zero.

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 masks 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 found four 17-round impossible differentials as follows.

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

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

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

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

We search these impossible differentials with Gurobi6.0.4. 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). Although we can not figure out the conflicts in these impossible differentials, but they are surely existed in theory.

4.3 17-Round Zero-Correlation Linear Approximations of HIGHT

Until now, for the HIGHT block cipher, the longest zero-correlation linear approximation is 16 rounds presented by Wen *et al.* in [31], 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 bit masks on the highest position of each branch of input and output, and found 4×128 zero-correlation linear approximations described as follows:

Property 1 (16-round zero-correlation linear trails of HIGHT in [31]). Set the input mask and output mask after 16 round HIGHT be $\alpha = (\alpha_7, \alpha_6, \ldots, \alpha_0)$ and $\beta = (\beta_7, \beta_6, \ldots, \beta_0)$ respectively. For any $\alpha_i = 00000001, \alpha_j = 0, j \neq i, 0 \leq i, j \leq 7, \beta_k = 1???????, \beta_l = 0, l \neq k, 0 \leq l, k \leq 7$, if $(i, k) \in \{(6,5), (4,3), (2,1), (0,7)\}$, then the correlation of this linear approximation $\alpha \xrightarrow{16r} \beta$ is zero, and for each $(i, k) \in \{(6,5), (4,3), (2,1), (0,7)\}$, there are 128 such linear approximations.

In this part we utilize the MILP models proposed in section 3 to search longer zero-correlation linear approximations for HIGHT cipher. 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.

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

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

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

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

We search these zero-correlation linear approximations with Gurobi6.0.4. 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).

5 Application to PRESENT

5.1 Brief Description of PRESENT

PRESENT cipher, proposed by Bogdanov *et al.* at CHES 2007 [6], is a lightweight block cipher designed for hardware constrained environments such as RFID tags and sensor networks. It adopts SP-network and consists of 31 rounds. Its block size is 64 bits and two key sizes of 80 and 128 bits are supported. The round function of PRESENT includes three layers. The first layer is a bitwise XOR operation to introduce a subkey $sk_i, 1 \leq i \leq 32$ into i-th round, where sk_{32} is used for post whitening. The second layer is a non-linear layer which is composed by 16 parallel 4×4 S-boxes. The third layer is a linear bitwise permutation. This round function is shown in Figure 2.

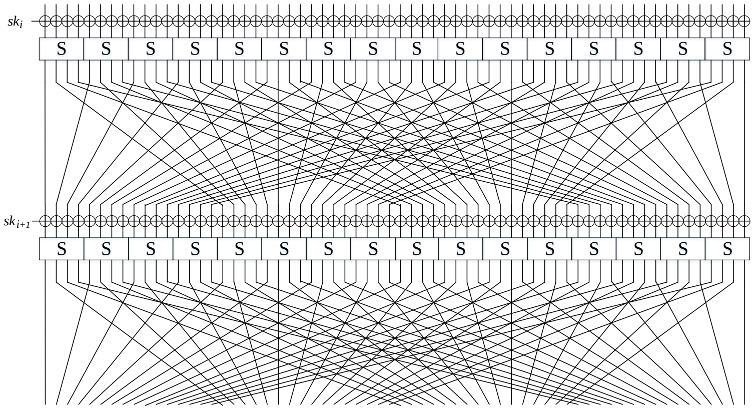
sk_i
S S S S S S S S S S S S S S S S
sk_{i+1}
S S S S S S S S S S S S S S S S

Fig. 2. 2-round PRESENT cipher

Since the key schedule is not related to the search of impossible differentials, we omit it in this paper. For further details, please refer to [6].

5.2 6-Round Impossible Differentials of PRESENT

Impossible differential cryptanalysis utilizes the differential with probability exactly zero. For PRESENT cipher, the longest impossible differential is proposed by Tezcan in [29] so far. He used the undisturbed bits, which is defined as a part bits of output (input) difference that happens with probability one when the input (output) difference are given, to find out a 6-round impossible differential. However, with existing widely applicable automatic search tools such as $\mathcal{U}$-method, UID-method and their improved method in [33], it is only possible to obtain 4-round impossible differentials for PRESENT ciphers, because they don't consider about the differential property of S-box.

In this section, we use the MILP-based automatic tool to search the impossible differentials for PRESENT cipher. Due to the limit of computing resource, we search the cases that the active bits of input difference and the intermediate state after the S-box layer of last round both happen only before one of 16 S-boxes, which means we only search $16 \times 16 \times (2^4 - 1) \times (2^4 - 1) = 57600$ cases. By exploiting this automatic search tool, we total find 5050 impossible differentials for 6-round PRESENT under the constraints above. We denote

$(0, \dots, 0, c_i, 0, \dots, 0) \nrightarrow (0, \dots, 0, c_j, 0, \dots, 0)$ as a 6-round impossible differential for PRESENT cipher, where the i-th nibble of input difference is c_i, the j nibble of the state difference after the last S-box layer is c_j, the differences on other nibbles are zero, then we classify all impossible differentials into several categories which are shown in table 1, each impossible differential belongs to at least one category.

i	c_i	j	c_j
$\in (0, 1, \dots, 15)$	9	$\in (1, 3, 5, 7, 9, 11, 13, 15)$	$\in (1, 2, \dots, 15)$
$\in (0, 1, 2, 3, 12, 13, 14, 15)$	$\in (1, 2, \dots, 15)$	$\in (0, 1, \dots, 15)$	5
$\in (12, 15)$	$\in (1, 2, \dots, 15)$	$\in (0, 1, 3, 4, 5, 7, 9, 11, 12, 13, 15)$	$\in (9, 13)$
$\in (12, 15)$	$\in (1, 3, 9, 11, 13, 15)$	2	$\in (9, 13)$
$\in (12, 15)$	$\in (1, 3, 8, 9, 11, 13, 15)$	$\in (6, 14)$	$\in (9, 13)$
$\in (12, 15)$	$\in (1, 3, 8, 11, 13)$	8	$\in (9, 13)$
$\in (12, 15)$	$\in (1, 8, 11, 13)$	10	$\in (9, 13)$
$\in (13, 14)$	$\in (1, 2, \dots, 15)$	$\in (1, 3, 4, 5, 7, 12, 13, 15)$	$\in (9, 13)$
$\in (13, 14)$	$\in (1, 3, 11, 13)$	0	$\in (9, 13)$
$\in (13, 14)$	$\in (1, 3, 13)$	2	$\in (9, 13)$
$\in (13, 14)$	$\in (1, 3, 9, 11, 13)$	$\in (6, 14)$	$\in (9, 13)$
15	8	2	$\in (9, 13)$
15	7	8	$\in (9, 13)$
15	3	10	$\in (9, 13)$

Table 1. All impossible differentials for 6-round PRESENT

Taking the first entry of table 1 as an illustration, if the input difference is zero on most nibbles except the i-th nibble, $i \in (0, 1, \dots, 15)$, and the state difference after the last S-box layer is zero on majority of nibbles except the j-th nibble, $j \in (1, 3, 5, 7, 9, 11, 13, 15)$, $(0, \dots, 0, c_i, 0, \dots, 0) \nrightarrow (0, \dots, 0, c_j, 0, \dots, 0)$ is an impossible differential when c_i is fixed to be 9,and c_j belongs to one of the set $(1, 2, \dots, 15)$. Note that there are some repeated impossible differentials in different entries.

We search these impossible differentials in table 1 with Gurobi6.0.4. 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).

6 Application to the existence proof

Since our new automatic search tool takes non-linear components such as S-box and modular addition into consideration, we can use this tool on the existence proof of impossible differentials and zero-correlation linear approximations for certain rounds of block ciphers by traversing all input and output differences (masks). However, the block size of a cipher is usually too large to deal with due to the time complexity, we generally search impossible differentials (zero-correlation linear approximations) from a subset of all cases that the input and output differences (masks) satisfy special patterns. Thus we can proof that in this particular subset whether there are impossible differentials and zero-correlation linear approximations or not. As instances, we can proof that the longest impossible differentials for LBlock, TWINE and Piccolo ciphers are 14, 14 and 7 rounds respectively when we only consider the patterns that the 8 nibbles of input difference which will enter the first round function are all zero and within the other 8 nibbles only one is non-zero difference. The requirement on the output difference is as same as that on input, which means we only focus on $(8 \times (2^4 - 1))^2 = 14400$ cases that satisfy the special patterns. What's more, by setting the hamming weights of both input and output differences (masks) are one, we find the longest impossible differentials (zero-correlation linear approximations) are 15, 15 rounds for TEA and XTEA ciphers respectively, and find the longest zero-correlation linear approximations are all 6 rounds for SPECK-32 and SPECK-48 ciphers. We believe that this tool can be used to the existence proof for most lightweight ciphers with S-box and ARX ciphers under special patterns.

7 Conclusion

In this paper, we propose an automatic search tool for impossible differentials and zero-correlation linear approximations based on mixed integer linear programming method. In this tool, the differential and linear properties of nonlinear components are taken into consideration, so we can find longer impossible differentials and zero-correlation linear approximations comparing with previous search methods for a target cipher. As applications, we apply this tool on HIGHT, PRESENT and the existence proof. As a result, we find 4 impossible differentials and 4 zero-correlation linear approximations for 17-round HIGHT, which are the longest ones for HIGHT cipher. For Sbox-based ciphers, as an application we find 5050 impossible differentials for 6-round PRESENT cipher. Actually, since in MILP model, 8 × 8 S-box cannot be exactly described with a set of inequalities, which is a limitation of our automatic search tool, in the further we will still research how to apply this tool on ciphers with large size of S-box.

References

1. Biham, E., Biryukov, A., Shamir, A. (1999, May). 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.
2. Biham, E., Shamir, A. (2012). Differential cryptanalysis of the data encryption standard. Springer Science Business Media.
3. Biham, E. (1994, May). On Matsui's linear cryptanalysis. In Workshop on the Theory and Application of of Cryptographic Techniques (pp. 341-355). Springer Berlin Heidelberg.

4. Biham, E., Biryukov, A., Shamir, A. (1999, March). Miss in the Middle Attacks on IDEA and Khufu. In International Workshop on Fast Software Encryption (pp. 124-138). Springer Berlin Heidelberg.
5. Blondeau, C. (2015). Impossible differential attack on 13-round Camellia-192. Information Processing Letters, 115(9), 660-666.
6. Bogdanov, A., Knudsen, L.R., Leander, G., Paar, C., Poschmann, A., Robshaw, M.J.B., Seurin, Y., Vikkelsoe, C.: PRESENT: An Ultra-Lightweight Block Cipher. In: Paillier, P., Verbauwhede, I. (eds.) CHES 2007. LNCS, vol. 4727, pp.450-466. Springer, Heidelberg (2007)
7. Bogdanov, A., Rijmen, V. (2014). Linear hulls with correlation zero and linear cryptanalysis of block ciphers. Designs, codes and cryptography, 70(3), 369-383.
8. Bogdanov, A., Wang, M. (2012). Zero correlation linear cryptanalysis with reduced data complexity. In Fast Software Encryption (pp. 29-48). Springer Berlin Heidelberg.
9. Bogdanov, A., Leander, G., Nyberg, K., Wang, M. (2012, December). 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.
10. Bogdanov, A., Geng, H., Wang, M., Wen, L., Collard, B. (2013, August). 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.
11. Bogdanov, A., Wang, M. (2012). Zero correlation linear cryptanalysis with reduced data complexity. In Fast Software Encryption (pp. 29-48). Springer Berlin Heidelberg.
12. Boura, C., Naya-Plasencia, M., Suder, V. (2014, December). 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.
13. Fu, K., Wang, M., Guo, Y., Sun, S., Hu, L. (2016). MILP-Based Automatic Search Algorithms for Diff erential and Linear Trails for Speck. representations, 21, 27.
14. Hong, D., Sung, J., Hong, S., Lim, J., Lee, S., Koo, B. S., ... Kim, H. (2006, October). 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.
15. ISO/IEC 18033-3, Information technology-Security techniques-Encryption algorithms-Part 3: Block ciphers (2010)
16. Kim, J., Hong, S., Lim, J. (2010). Impossible differential cryptanalysis using matrix method. Discrete Mathematics, 310(5), 988-1002.
17. Knudsen, L. (1998). DEAL-a 128-bit block cipher. complexity, 258(2), 216.
18. Lipmaa, H., Moriai, S. (2001, April). Efficient algorithms for computing differential properties of addition. In International Workshop on Fast Software Encryption (pp. 336-350). Springer Berlin Heidelberg.
19. Lu, J. (2007, November). 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.
20. Luo, Y., Lai, X., Wu, Z., Gong, G. (2014). A unified method for finding impossible differentials of block cipher structures. Information Sciences, 263, 211-220.
21. Mala, H., Dakhilalian, M., Rijmen, V., Modarres-Hashemi, M. (2010, December). Improved impossible differential cryptanalysis of 7-round AES-128. In Interna-

tional Conference on Cryptology in India (pp. 282-291). Springer Berlin Heidelberg.

22. Matsui, M. (1993, May). Linear cryptanalysis method for DES cipher. In Workshop on the Theory and Application of of Cryptographic Techniques (pp. 386-397). Springer Berlin Heidelberg.
23. Mouha, N., Wang, Q., Gu, D., Preneel, B. (2011, November). Differential and linear cryptanalysis using mixed-integer linear programming. In International Conference on Information Security and Cryptology (pp. 57-76). Springer Berlin Heidelberg.
24. Nyberg, K., Wallén, J. (2006, March). Improved linear distinguishers for SNOW 2.0. In International Workshop on Fast Software Encryption (pp. 144-162). Springer Berlin Heidelberg.
25. Sun, B., Liu, Z., Rijmen, V., Li, R., Cheng, L., Wang, Q., ... Li, C. (2015, August). Links among impossible differential, integral and zero correlation linear cryptanalysis. In Annual Cryptology Conference (pp. 95-115). Springer Berlin Heidelberg.
26. Sun, S., Hu, L., Wang, P., Qiao, K., Ma, X., Song, L. (2014, December). 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.
27. Sun, S., Hu, L., Song, L., Xie, Y., Wang, P. (2013, November). 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.
28. Sun, S., Hu, L., Wang, M., Wang, P., Qiao, K., Ma, X., ... Fu, K. (2014). 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.
29. Tezcan, C. (2014). Improbable differential attacks on Present using undisturbed bits. Journal of Computational and applied mathematics, 259, 503-511.
30. Wallén, J. (2003, February). Linear approximations of addition modulo 2^n. In International Workshop on Fast Software Encryption (pp. 261-273). Springer Berlin Heidelberg.
31. Wen, L., Wang, M., Bogdanov, A., Chen, H. (2014). Multidimensional zero-correlation attacks on lightweight block cipher HIGHT: Improved cryptanalysis of an ISO standard. Information Processing Letters, 114(6), 322-330.
32. Wu, S., Wang, M. (2011). Security Evaluation against Differential Cryptanalysis for Block Cipher Structures. IACR Cryptology ePrint Archive, 2011, 551.
33. Wu, S., Wang, M. (2012, December). Automatic search of truncated impossible differentials for word-oriented block ciphers. In International Conference on Cryptology in India (pp. 283-302). Springer Berlin Heidelberg.