IET Information Security

Research Article

IET Journals
The Institution of Engineering and Technology

MILP-based automatic differential search for LEA and HIGHT block ciphers

ISSN 1751-8709
Received on 22nd October 2018
Revised 28th March 2020
Accepted on 15th May 2020
E-First on 6th July 2020
doi: 10.1049/iet-ifs.2018.5539
www.ietdl.org

Elnaz Bagherzadeh[1], Zahra Ahmadian[1] ✉

[1]*Department of Electrical Engineering, Shahid Beheshti University, Tehran, Iran*

✉ *E-mail: zahmadian@sbu.ac.ir*

Abstract: The authors use the mixed-integer linear programming (MILP) technique for the automatic search for differential characteristics of LEA and HIGHT ciphers. They show that the MILP model of the differential property of modular addition with one constant input can be represented with a much lesser number of linear inequalities compared to the general case. Benefiting from this model for HIGHT block cipher, they can achieve a reduction of $112r$ out of $480r$ in the total number of linear constraints for the MILP model of r-round of HIGHT. This saving accelerates the searching process of HIGHT about twice as fast. They enjoy the MILP model to investigate the differential effect of these ciphers and provide a more accurate estimation for the differential probability. Their observations show that despite HIGHT, LEA exhibits a strong differential effect. The results gained by this method improve/extend the previous results as follows. For LEA block cipher, they found more efficient 12- and 13-round differentials whose probabilities are better than the best previous 12- and 13-round differentials for a factor of about 2^6 and 2^7, respectively. In the case of HIGHT block cipher, they found new 12- and 13-round differentials, though with the same best-reported probabilities.

Nomenclature

n	word size in bits
$x[i]$	the ith bit of the n-bit word x, $0 \le i < n$, $x[0]$ = least significant bit, $x[n-1]$ = most significant bit
$\oplus$	bitwise logical exclusive OR (XOR)
$+$	modular addition in modules 2^n.
$<<< i$	left cyclic rotation by i bits
Δ_{in}^*	input difference of the optimum differential characteristic
Δ_{out}^*	output difference of the optimum differential characteristic
Δ_i^*	difference of the optimum differential characteristic in round i
$xdp^+(\alpha, \beta \to \gamma)$	differential probability of addition modulo 2^n with input differences α, β and output difference γ
$p(x)$	probability polynomial

1 Introduction

In symmetric cryptography, ARX structure refers to designs in which only three operations are used: modular addition, XOR, and the rotation. This strategy is regarded as an important alternative for substitution permutation network (SPN) structure and contributes a large portion of the existing symmetric schemes. For example, two secure hash algorithm-3 finalists BLAKE [1] and Skein [2] hash functions, the eStream finalist Salsa20 [3], and the national security agancy released block cipher SPECK [4] are some well-known instances of ARX structure. Some other examples are SipHash hash function [5], Chaskey message authentication code algorithm [6], ChaCha [7], HC-128 [8], stream ciphers and LEA [9], Threefish [2], RC5 [10], and HIGHT [11] lightweight block ciphers.

An important step in designing any symmetric scheme is to evaluate its resistance against differential attack. Despite SPN ciphers which enjoy some provable upper bounds for the probability of differential characteristics, such a feature for the ARX structures has not yet been found. Therefore, it has been a focus of cryptographers' concerns to automise the algorithms of searching the optimal differential characteristics. In this regard, a variety of methods have been proposed and applied such as the methods adopted from the branch and bound Matsui's algorithm [12, 13], the methods based on Boolean satisfiability problems [14–16], and the methods based on mixed-integer linear programming (MILP) problems [17, 18].

In this study, we focus on the third technique which has been explicitly applied for automated search algorithms for cryptanalysis of symmetric ciphers either SPN or ARX structures [17–25]. MILP is a class of optimisation problems derived from linear programming (LP), which aims to optimise an objective function under a certain set of constraints. Although this problem is non-polynomial (NP)-complete inherently, there are some open-source and commercially available MILP solvers that can solve some MILP problem instances, which are not too large and complicated. To employ this tool for cryptanalysis, the problem of cryptanalysis of a symmetric cipher should be translated into a MILP problem and then be solved by a MILP solver such as solving constraint integer programs (SCIP) optimiser [26]. The initial attempts for employing the MILP technique for cryptanalysis of symmetric ciphers belong to the SPN ciphers where Mouha *et al.* [19] and Wu and Wang [20] converted the problem of finding the minimum number of active S-boxes into a MILP problem. This method was then used for finding some specific pattern characteristics for AES-based lightweight authenticated encryption authenticated encryption algorithm and counting the minimal number of active S-boxes of bit-oriented block ciphers by introducing bit-level representations in [27]. Sun *et al.* [23] extended Mouha *et al.* method to analyse block ciphers with bitwise permutation diffusion layers (S-bP structures) in the single key and related key models, though without considering the differential properties of S-box modules. In [21], the differential properties of the S-box layer have been taken into account in the MILP model and more precise results for differential characteristics were derived, consequently.

The main challenge in the case of the ARX ciphers is to construct an efficient MILP model to represent the differential pattern of modular addition. Although the algorithm proposed in [23] has shown to be effective in constructing MILP models for (at most 4×4) S-box modules, it cannot be used for modular addition such as a $2n \times n$ S-box in $\mathbb{F}_2^n$ (for n typically at least 16). Since it

Table 1 Comparison of the number of constraints for MILP models for r rounds of HIGHT

Model	Number of constraints
original 1	$776r$
original 2	$480r$
Yin *et al.* [29]	$694r$
this paper	$368r$

Table 2 Comparison of our characteristics of LEA with previous ones

Rounds	Characteristic prob.	Differential prob.	Reference
11	2^{-98}	$2^{-91.9}$	[9]
12	2^{-128}	—	[9]
12	2^{-112}	$2^{-101.71}$	[30]
12	2^{-107}	$2^{-95.86}$	this paper
13	2^{-134}	$2^{-123.02}$	[30]
13	2^{-127}	$2^{-115.86}$	this paper

The bold row distinguishes the contribution of this work from other works.

Table 3 Comparison of our characteristics of HIGHT with the previous one

Rounds	Characteristic prob.	Differential prob.[a]	Reference
11	2^{-58}	2^{-58}	[11]
11	2^{-45}	2^{-45}	[29] and this paper
12	2^{-53}	2^{-53}	[29] and this paper
13	2^{-61}	2^{-61}	[29] and this paper

[a][29] does not provide any analysis for differential probability.

The bold row distinguishes the contribution of this work from other works.

demands too many linear constraints, which make the MILP problem of a typical ARX cipher too complex and hence intractable.

Fu *et al.* [17] resolved this problem by utilising the differential property of modular addition provided in [28]. They derived an efficient MILP model for modular addition and applied it to SPECK ARX cipher and improved the existing results.

In this study, we modify and use the MILP model for ARX structures proposed in [17] to find differential characteristics for ARX ciphers LEA and HIGHT. The design of HIGHT involves some modular additions whose one input is constant (i.e. with zero difference). Although such a scenario does not violate the general form analysed in [17], it could be modelled much more efficiently with lesser number of constraints if it is revisited as a new problem. We did this revision and reduced the number of constraints from $13n + 1$ into $5n + 1$ per modular addition, where n is the word size. This improvement reduced the number of constraints from $480r$ into $368r$ for r-round HIGHT, which is a considerable improvement and makes the search process of HIGHT about twice as fast. Recently, HIGHT has received another differential characteristic search using the MILP method [29] with a number of $694r$ constraints for r-round HIGHT. Using this model, new 12- and 13-round characteristics were proposed for HIGHT [29]. Table 1 compares the number of MILP constraints for the proposed model and other models. By original models 1 and 2, we mean the MILP model in which the number of XOR constraints is five and one, respectively.

Moreover, we compute the probability of the (sub-)optimal differential, rather than the (sub-)optimal characteristic only. The notion *probability polynomial* is defined to reflect the differential effect for each cipher in a compact form. Our results show that despite HIGHT, LEA exhibits a strong differential effect, which makes the differential probability much higher than its dominant characteristic probability. A summary of our achievements along with the previous results for LEA and HIGHT are shown in Table 2 and 3, respectively. For LEA, we found new 12- and 13-round differentials with improved probabilities. For HIGHT, using our new model, we found new differentials for 11, 12, and 13 rounds, apart from those introduced in [29].

The rest of this paper is organised as follows. Section 2 describes the MILP model for differential characteristics in ARX ciphers, where Fu *et al.* model along with our proposed model for a special case of modular addition is presented. In Section 3, we reviewed the concepts of characteristic and differential and the new concept of probability polynomial is presented. Our results on LEA and HIGHT ciphers are detailed in Sections 4 and 5, respectively. Finally, Section 6 concludes our work.

2 MILP model for differential characteristics in ARX ciphers

The MILP problem is the problem of optimising the value of a linear objective function of some integer/real-valued variables, which satisfy some linear (in)equality constraints.

MILP solvers can be enjoyed to find the best differential characteristic of a cipher if the problem of finding the optimal differential characteristic of the cipher can be translated into a (not-too-complex) MILP problem. To that end, the objective function should be set equal to an adequate strictly monotonic function of the characteristic probability, and the linear constraints are configured in such a way that they express the propagation of the difference values in the cipher. Therefore, with respect to the modelled cryptosystem, the optimum differential characteristic probability would be returned by solving the model by an adequate MILP solver.

Fu *et al.* [17] proposed the first MILP model for the differential characteristic search problem of ARX structures by defining the objective function as well as the linear constraints for ARX structures.

Among the set of modules involved in ARX structures, rotation operation only changes the position of the input bits, so a simple change of variables describes the input difference–output difference relation completely. Since rotation is a linear operation, it does not contribute to the characteristic probability and consequently the objective function. In this section, we first review the MILP model for differential properties of the XOR and modular addition operations [17], then we propose the new and more concise MILP model for the modular addition when one of its inputs is constant. The notation used in the paper is summarised in Nomenclature section.

2.1 MILP model for XOR

For the XOR operation with bit-level input and output differences, Sun *et al.* [21] proposed a model including five inequalities in three input/output binary variables and an extra dummy binary variable that precisely describes the XOR operation. However, considering that all variables in the model are binary-valued, it was shown in [18] that the following single linear equation completely describes the XOR operation:

$$a + b + c = 2d \tag{1}$$

where d is a dummy binary variable. Since XOR is a linear operation, it does not have any effect on the characteristic probability and hence the objective function.

In [29], Yin *et al.* tried to improve the MILP model for two consecutive XORs, which is equivalent to a three-input XOR. Without noticing (1), they improved the originally 10-constraint model to an 8-constraint model. However, using (1), two consecutive XORs can be modelled by using two constraints only.

2.2 MILP model for modular addition

Based on the two following theorems derived by Lipmaa and Moriai [28], all the feasible differential characteristics for modular addition and their corresponding probabilities are characterised completely. In the following, the notation $x[i]$ is used to show the

ith bit of n-bit word x where the least significant bit and most significant bit of x are $x[0]$ and $x[n-1]$, respectively.

Theorem 1: The differential $(\alpha, \beta \to \gamma)$ is possible if the following two conditions are satisfied:

(i) $\alpha[0] \oplus \beta[0] \oplus \gamma[0] = 0$ and
(ii) if $\alpha[i-1] = \beta[i-1] = \gamma[i-1]$, then $\alpha[i-1] = \beta[i-1] = \gamma[i-1] = \alpha[i] \oplus \beta[i] \oplus \gamma[i]$, where $i = 1, \ldots, n-1$.

Theorem 2: Assume that $(\alpha, \beta \to \gamma)$ is a possible differential characteristic, then the XOR differential probability of addition (xdp^+) of this differential is

$$xdp^+(\alpha, \beta \to \gamma) = 2^{-\sum_{i=0}^{n-2} \neg\mathrm{eq}(\alpha[i], \beta[i], \gamma[i])} \tag{2}$$

where

$$\mathrm{eq}(\alpha[i], \beta[i], \gamma[i]) = \begin{cases} 1 & \alpha[i] = \beta[i] = \gamma[i] \\ 0 & \text{o . w} \end{cases} \tag{3}$$

The first feasibility condition $\alpha[0] \oplus \beta[0] \oplus \gamma[0] = 0$ in Theorem 1 can be described in the MILP model by the following equation:

$$\alpha[0] + \beta[0] + \gamma[0] = 2d$$

where d is a dummy binary variable.

In [17], Fu *et al*. observed that the second feasibility condition of Theorem 1 is equivalent to the fact that there are 56 possible vectors of the form

$$\begin{gathered}(\alpha[i], \beta[i], \gamma[i], \alpha[i+1], \beta[i+1], \gamma[i+1], \\ \neg\mathrm{eq}(\alpha[i], \beta[i], \gamma[i])\end{gathered} \tag{4}$$

in total. The SAGE Computer Algebra System [31] returns a set of 65 linear inequalities satisfying all these 56 possible patterns. However, 65 inequalities are too many, which makes the MILP model too complicated. In [17], it has been shown that based on the greedy algorithm given in [22], the number of linear inequalities can be reduced from 56 to 13. These 13 inequalities are shown in Table 4 [17].

Therefore, for n-bit words, the total number of constraints describing the addition module is $13(n-1)+1$.

The only non-linear module in the ARX structure is modular addition. So, it is the only contributor to the objective function. If there are r modular additions in the cipher in total, with input–output differences $(\alpha_j, \beta_j \to \gamma_j)$, $j = 1, \ldots, r$, then according to Theorem 2, the overall characteristic probability is

$$P_D = 2^{-\sum_{j=1}^{r} \sum_{i=0}^{n-2} \neg\mathrm{eq}(\alpha[i], \beta[i], \gamma[i])} \tag{5}$$

So, the objective function can be defined as

$$\sum_{j=1}^{r} \sum_{i=0}^{n-2} \neg\mathrm{eq}\big(\alpha_j[i], \beta_j[i], \gamma_j[i]\big) \tag{6}$$

which is a linear function and supposed to be minimised.

2.3 MILP model for modular addition with a constant input

Suppose that there is a modular addition in the cipher whose one input is constant. In other words, its corresponding difference is zero. This is exactly the case with HIGHT cipher, where some subkeys are added to the data in each round. One approach to handle this situation is to use the 13 general inequalities given in Table 4 directly, in which one of its input differences, say α, has been set to zero, i.e. $\alpha[i] = 0$, $i = 0, \ldots, n-1$.

Table 4 Linear inequalities expressing the differential property of modular addition in the general form [17]

$\beta[i] - \gamma[i] + (\neg\mathrm{eq}(\alpha[i], \beta[i], \gamma[i]))$	$\geq$	0,
$\alpha[i] - \beta[i] + (\neg\mathrm{eq}(\alpha[i], \beta[i], \gamma[i]))$	$\geq$	0,
$-\alpha[i] + \gamma[i] + (\neg\mathrm{eq}(\alpha[i], \beta[i], \gamma[i]))$	$\geq$	0,
$-\alpha[i] - \beta[i] - \gamma[i] - (\neg\mathrm{eq}(\alpha[i], \beta[i], \gamma[i]))$	$\geq$	−3,
$\alpha[i] + \beta[i] + \gamma[i] - (\neg\mathrm{eq}(\alpha[i], \beta[i], \gamma[i]))$	$\geq$	0,
$-\beta[i] + \alpha[i+1] + \beta[i+1] + \gamma[i+1] + (\neg\mathrm{eq}(\alpha[i], \beta[i], \gamma[i]))$	$\geq$	0,
$\beta[i] + \alpha[i+1] - \beta[i+1] + \gamma[i+1] + (\neg\mathrm{eq}(\alpha[i], \beta[i], \gamma[i]))$	$\geq$	0,
$\beta[i] - \alpha[i+1] + \beta[i+1] + \gamma[i+1] + (\neg\mathrm{eq}(\alpha[i], \beta[i], \gamma[i]))$	$\geq$	0,
$\alpha[i] + \alpha[i+1] + \beta[i+1] - \gamma[i+1] + (\neg\mathrm{eq}(\alpha[i], \beta[i], \gamma[i]))$	$\geq$	0,
$\gamma[i] - \alpha[i+1] - \beta[i+1] - \gamma[i+1] + (\neg\mathrm{eq}(\alpha[i], \beta[i], \gamma[i]))$	$\geq$	−2,
$-\beta[i] + \alpha[i+1] - \beta[i+1] - \gamma[i+1] + (\neg eq(\alpha[i], \beta[i], \gamma[i]))$	$\geq$	−2,
$-\beta[i] - \alpha[i+1] + \beta[i+1] - \gamma[i+1] + (\neg\mathrm{eq}(\alpha[i], \beta[i], \gamma[i]))$	$\geq$	−2,
$-\beta[i] - \alpha[i+1] - \beta[i+1] + \gamma[i+1] + (\neg\mathrm{eq}(\alpha[i], \beta[i], \gamma[i]))$	$\geq$	−2.

Table 5 Linear inequalities expressing the differential property of modular addition with a constant input

$-\beta[i] + (\neg\mathrm{eq}(\alpha[i], \beta[i], \gamma[i]))$	$\geq$	0,
$-\gamma[i] + (\neg\mathrm{eq}(\alpha[i], \beta[i], \gamma[i]))$	$\geq$	0,
$\beta[i] + \gamma[i] - (\neg\mathrm{eq}(\alpha[i], \beta[i], \gamma[i]))$	$\geq$	0,
$\beta[i+1] - \gamma[i+1] + (\neg\mathrm{eq}(\alpha[i], \beta[i], \gamma[i]))$	$\geq$	0,
$-\beta[i+1] + \gamma[i+1] + (\neg\mathrm{eq}(\alpha[i], \beta[i], \gamma[i]))$	$\geq$	0.

Here, we propose a more efficient model for this case with a much lesser number of inequalities. Again consider the 7-tuple vector given in (4). The first condition of Theorem 1 is simplified into $\beta[0] = \gamma[0]$, considering $\alpha[0] = 0$. This new form does not need defining any extra dummy variable, hence one bit saving in the number of variables of the problem.

To the second condition of Theorem 1, we add the new condition $\alpha[i] = \alpha[i-1] = 0$. So, the number of valid remaining vectors reduces to 14 possible patterns. Using the SAGE computer algebra system, we get ten linear inequalities satisfying all the 14 possible patterns and no impossible patterns. Then, we use the greedy algorithm to make this set smaller and finally the number of inequalities reduces from 10 to 5 inequalities (per bit), which is a great improvement in the number of constraints compared to 13 inequalities per bit, in the general case. In other words, for an n-bit modular addition, the number of constraints decreases by $8(n-1)$, where n is the word size. This set of inequalities is listed in Table 5. Obviously, the objective function does not change any.

Having defined the objective function and all the constraints for the target ARX cipher, the MILP model is complete and ready to be solved by a MILP solver. It worth mentioning that MILP solvers are capable of returning the number of distinct solutions along with the optimum value of the objective function.

3 Characteristic probability and differential probability using MILP method

To precisely evaluate the security of block ciphers against differential analysis Lai *et al*. first introduced the theory of Markov ciphers and made a distinction between a differential and a differential characteristic [32]. The only important factor in differential cryptanalysis is the values of input and output differences, no matter what the intermediate differences may be. However, for a given differential with fixed input–output differences, there could be potentially many characteristics that share the same input–output differences and so they all contribute to the magnitude of differential probability. Such an effect is called the strong differential effect [32]. To calculate the differential probability as accurately as possible, more characteristics sharing the same input and output differences should be counted in.

Therefore, in general, any differential will have a probability greater than that of its most probable characteristic. So, by considering the differential probability rather than the characteristic

Input: A raw MILP model for r rounds of the cipher
Output: Probability polynomial of the optimum characteristic

1 Solve the raw MILP model and obtain $(\Delta^*_{in}, \Delta^*_{out})$ for the optimum differential characteristic, along with d ;
2 $j \leftarrow -1$;
3 **repeat**
4 $\quad j \leftarrow j + 1$;
5 $\quad$ Set the input-output difference in the raw MILP model equal to $(\Delta^*_{in}, \Delta^*_{out})$;
6 $\quad$ Put the objective function in the raw MILP model equal to $d + j$;
7 $\quad$ Solve the model and $p_j \leftarrow$ number of answers ;
8 **until** $p_j 2^{-(d+j)} \geq \sum_{i=0}^{j-1} p_i 2^{d+i}$;
9 $N \leftarrow j - 1$;
10 **return** *probability polynomial*;

Fig. 1 *Algorithm 1: differential probability of the optimum characteristic*

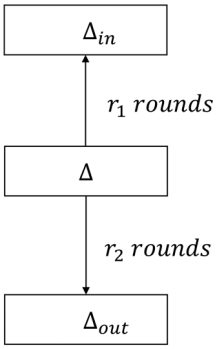

Fig. 2 *Obtaining a longer characteristic from two shorter ones*

probability, we calculate the true success rate of the differential cryptanalysis, not just a lower bound for that.

However, the MILP-based search tool finds only the most probable characteristic rather than differential. In the following, we explain how to employ the MILP model to find not only the best characteristic but also to compute the probability of the differential that matches this characteristic.

3.1 Computing differential probability

Assume that one has already constructed an MILP model for finding optimal differential characteristics, which we call the raw MILP model. This model has been solved by an MILP solver and the optimum characteristic has been found with an input–output difference $(\Delta^*_{\text{in}}, \Delta^*_{\text{out}})$ and probability 2^{-d}. It means that the objective function in the raw MILP model has an optimum value equal to d. Now, we explain how this raw model can be modified in such a way to compute the differential probability by finding other characteristics with the same input–output differences $(\Delta^*_{\text{in}}, \Delta^*_{\text{out}})$ with probabilities $\leq 2^{-d}$.

We first introduce the notion *probability polynomial* that we define for a compact and concise representation of the probability of a differential and its corresponding characteristics. The probability polynomial of a specific differential with a given input–output difference is defined as follows:

$$p(x) = p_0 x^d + p_1 x^{d+1} + p_2 x^{d+2} + \cdots \tag{7}$$

where 2^{-d} is the probability of its dominant characteristic and p_i is the number of distinct characteristics with the probability of $2^{-(d+i)}$, $i = 0, 1, \ldots$. It is clear that the probability of the corresponding characteristic can be calculated by evaluating $p(x)$ at $x = \frac{1}{2}$. In particular, we consider the truncated version of $p(x)$, containing only the first N monomials, i.e.

$$p(x) \simeq p_0 x^d + p_1 x^{d+1} + \cdots + p_N x^{d+N} \tag{8}$$

where N has been actually selected in such a way that at $x = 1/2$ the next term is negligible compared to the sum of the first N terms, i.e.

$$p_{N+1} 2^{-(d+N+1)} < \sum_{i=0}^{N} p_i 2^{-(d+i)} \tag{9}$$

Algorithm 1 (see Fig. 1) shows how to construct the truncated probability polynomial for the optimal characteristic, given the raw MILP model. Despite line 1 where the MILP solver is configured to return the optimum solution along with the values of variables, in line 7 it should be configured to return the number of optimum solutions.

3.2 Sub-optimal solutions

The MILP problem is inherently an NP-complete problem. So, it is not unexpected that the MILP solver fails to solve a complicated MILP model with a large number of variables and constraints corresponding to differential cryptanalysis.

In such problems, if the solver fails to solve the problem as a whole, a sub-optimal solution may suffice. To find a sub-optimal solution, it is very conventional to divide the r-round cipher into two r_1- and r_2-round subciphers ($r = r_1 + r_2$), and solve each problem independently [17, 30]. Definitely, the output difference of the first subcipher must be the same as the input difference of the second subcipher. So, the MILP models of the first r_1-round and second r_2-round subciphers must have an extra constraint, which is, respectively, the output difference $= \Delta$ and the input difference $= \Delta$. Finally, If the optimum value of the first and second problems are d_1 and d_2, respectively, the sub-optimum value for the full r-round problem would be $d = d_1 + d_2$. This process has been shown in Fig. 2. This is exactly equal to the main r-round problem, which is subjected to the extra constraint $\Delta_{r_1} = \Delta$.

The only thing that remains is to limit the set of candidate values for Δ to a small enough set with appropriate values. We should search this set of Δ and choose the one with the highest d value. The differential property of modular addition shows that the more active bits in the input–output differences, potentially the weaker probability of the differential. So, a common choice Δ is always a low-weight one, e.g. those with exactly a single active bit.

After finding the best Δ and the associated optimum values d_1 and d_2, we run Algorithm 1 for the two subciphers independently to construct the two probability polynomials $p_1(x)$ and $p_2(x)$. To derive the probability polynomial of the main r-round cipher, it is needed to consider all possible r-round characteristics by combining each r_1-round characteristic and each r_2-round characteristic. This process is exactly equivalent to multiplying the probability polynomials of the two subciphers. So, the probability polynomial of the main r-round differential would be

$$p(x) = p_1(x) p_2(x) \tag{10}$$

In general, the main problem may be so complex that dividing the r-round cipher into just two subciphers would not be sufficient. So, let the r-round cipher be divided into k subciphers with probability polynomials $p_i(x)$, $i = 1, \ldots, k$. Clearly, the output difference of subcipher i is equal to the input difference of subcipher $i + 1$. Finally, the probability polynomial of the r-round cipher is

$$p(x) = \prod_{i=1}^{k} p_i(x) \tag{11}$$

and the differential probability is $p(x)|_{x=\frac{1}{2}}$.

4 Differential analysis of LEA block cipher using MILP method

LEA is an ARX block cipher proposed by Hong *et al.* in Workshop on Information Security Applications 2009 [9]. It provides high-

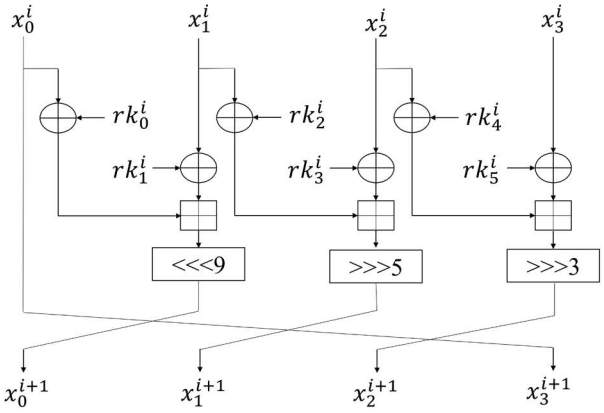

Fig. 3 *Round function of LEA*

Table 6 Sub-optimal characteristic for 12-round LEA

Rounds	12-round	
—	$\Delta x_0 \| \| \Delta x_1 \| \| \Delta x_2 \| \| \Delta x_3$	$\log_2 p$
0	0xC0000000C0400080404000010404000012	—
1	0x8001000080000004400000014400000010	−13
2	0x0200080082000000100000010010800	−8
3	0x0010010000100000000200002000800	−4
4	0x000200000001FF000040030000100100	−15
5	0x0002000000020000000600000020000	−25
6	**0x00000000000000000000000000020000**	**−5**
7	0x00000000000000000004000000000000	−1
8	0x00000000000060000000800000000000	−3
9	0x00040000000001000000100000000000	−6
10	0x08042000800000080000002000040000	−5
11	0x00401110C40000000000800408002000	−8
12	0x80222188222004008100140000401110	−14
	$\log_2(P_{\text{char}}) = \sum_r \log_2(p_r)$	−107
	$\log_2(P_{\text{diff}}) >$	−95.86

The bold row is the middle row of the cipher, from which the cipher is divided into two parts.

speed software encryption on general-purpose processors. It has a block size of 128 bits and a key size of 128, 192, or 256 bits. There are some cryptanalyses on LEA in the literature such as [30, 33].

In [9], the designers of LEA proposed the first differential analysis of their scheme. Their analysis is confined to finding characteristic probability and not a differential probability. The characteristic under consideration has been found by linearising LEA (replacing modular addition with XOR) and conditioned that the Hamming weight of the difference in the middle of the cipher is small. So, their best findings are 11-round and 12-round characteristics with probabilities 2^{-98} and 2^{-128}, respectively. Song *et al.* [30] used a search method based on SAT solvers and found characteristics and differentials for 12 rounds and 13 rounds of LEA with probability better than 2^{-128}. There is also an informally published work on 12-round (This work is incorrectly reported as a 13-round characteristic in [31].) LEA with probability 2^{-121} [34] using a search method based on the nested Monte–Carlo algorithm. In this section, we report our MILP-based results which outperform the previous ones [9, 30]. All results on LEA have been summarised in Table 2.

4.1 LEA specification

The encryption algorithm of LEA works as follows. It maps the plaintext of four 32-bit words $(x_0^0, x_1^0, x_2^0, x_3^0)$ into the ciphertext $(x_0^r, x_1^r, x_2^r, x_3^r)$ using a sequence of operations for r rounds, where $r = 24$ for LEA-128, $r = 28$ for LEA-192, and $r = 32$ for LEA-256. The round function for round i, $i = 0, \ldots, r-1$ is defined as follows:

$$\begin{aligned} x_0^{i+1} &\leftarrow ((x_0^i \oplus rk_0^i) + ((x_1^i \oplus rk_1^i)) < < < 9 \\ x_1^{i+1} &\leftarrow ((x_1^i \oplus rk_2^i) + ((x_2^i \oplus rk_3^i)) > > > 5 \\ x_2^{i+1} &\leftarrow ((x_2^i \oplus rk_4^i) + ((x_3^i \oplus rk_5^i)) > > > 3 \\ x_3^{i+1} &\leftarrow x_0^i. \end{aligned} \tag{12}$$

One round of LEA cipher has been shown in Fig. 3.

4.2 MILP-based search for characteristics and differentials of LEA

According the MILP model for differential attack on ARX structures described in Section 2, we can construct an MILP model for one-round and hence any arbitrary rounds of LEA cipher. All the XOR operations in LEA are used for key addition, which are bypassed in the differential attack. So, for each round of LEA, our model includes three modular additions and a bit permutation as for the rotation and words swapping. Therefore, the total number of constraints would be $3(13(n-1)+1)+4n$, where the word size in LEA is $n = 32$. So, the total number of constraints for each round of LEA becomes 404.

However, to search an r-round LEA without any extra constraint, the MILP model will become too complex to be solved for $r \geq 4$. Therefore, according to the discussion in Section 3.2, we choose the strategy of finding a sub-optimal solution and construct a long characteristic from two short ones. All the differential characteristics in this study have been searched out with the non-commercial MILP solver SCIP [26], on a personal computer (Intel(R) Core(TM) i7-4500U CPU@1.80 GHz 2.40 GHz, 10.0 GB RAM).

4.2.1 Analysis of 12-round LEA: To this end, we first analyse $r = 12$ rounds of LEA by dividing it into two subciphers of $r_1 = r_2 = 6$ rounds. The first subcipher has exactly one active bit, say bit i, in its output difference and the second one has the same pattern in its input difference. The two problems are solved independently and optimum values d_1 and d_2 are derived for $i = 0, \ldots, 127$. Among all 128 possible cases, the sub-optimal characteristic for 12-round LEA is that with the minimum $d = d_1 + d_2$. So, in this way, we found a 12-round characteristic for LEA with the additional constraint that its internal difference at round 6 has Hamming weight equal to one.

A six-round MILP problem for LEA, which is constrained to have Hamming weight one either in input or in output, is fortunately solvable by our machine. Among all 128 possible cases, the best one occurs at $i = 110$ which is equal to a 12-round characteristic with internal difference $\Delta_6^* = (0x00000000, 0x00000000, 0x00000000, 0x00020000)$. For this case, $d_1 = 70$ and $d_2 = 37$. So, the corresponding sub-optimum 12-round characteristic has $d = 107$. The details of this characteristic are shown in Table 6.

To find the differential probability corresponding to the sub-optimum characteristic, we first find probability polynomials $p_1(x)$ and $p_2(x)$ according to Algorithm 1

$$\begin{aligned} p_1(x) = {} & 3x^{70} + 9x^{71} + 32x^{72} + 101x^{73} + 245x^{74} \\ & + 635x^{75} + 1462x^{76} + 3107x^{77} + 5264x^{78}, \end{aligned} \tag{13}$$

$$\begin{aligned} p_2(x) = {} & 2x^{37} + 0x^{38} + 10x^{39} + 15x^{40} + 24x^{41} \\ & + 70x^{42} + 112x^{43} + 254x^{44} + 505x^{45} + 731x^{46}. \end{aligned} \tag{14}$$

Now we obtain the probability polynomial for 12-round differential as follows;

$$
\begin{aligned}
p(x) &= \prod_{i=1}^{2} p_i(x) = p_1(x)p_2(x) \\
&= 6x^{107} + 18x^{108} + 94x^{109} + 337x^{110} \\
&\quad + 1017x^{111} + 3186x^{112} + 8623x^{113} \\
&\quad + 22673x^{114} + 55008x^{115} + 111568x^{116} \\
&\quad + 254616x^{117} + 463615x^{118} + 866416x^{119} \\
&\quad + 1587582x^{120} + 2581241x^{121} + 3974813x^{122} \\
&\quad + 4929537x^{123} + 3847984x^{124}
\end{aligned} \tag{15}
$$

Finally, by evaluating $p(x)$ at $x = \frac{1}{2}$ we end up with the differential probability of 12-round LEA, which is

$$
p(x)\Big|_{x=\frac{1}{2}} = 2^{-95.8629} \tag{16}
$$

4.2.2 Analysis of 13-round LEA: To find a sub-optimum 13-round characteristic, we first examined the scenario of dividing it into 6-round and 7-round subciphers. However, the MILP problem for 7-round is not solvable by our machine, even when it is constrained to have weight one in the input or output. So, we have to divide the cipher into three subciphers. A good choice, though not necessarily the best one, is to continue the already found 12-round characteristic. So, in the case of the 13-round, the first two subciphers would be the two previously found 6-round subciphers and the third one would be a 1-round which is constrained such that its input difference is equal to the output difference of the second subcipher, i.e. $\Delta_{12}^{*} = (0x80222188, 0x22200400, 0x81001400, 0x00401110)$. This 1-round characteristic along with the two previous 6-rounds are reported in Table 7. Comparing Tables 6 and 7, one can realise that two distinct maximal probability characteristics are reported in these tables, but with the same Δ_1^{*} and Δ_{12}^{*}. The third subcipher probability polynomial $p_3(x)$ has only one monomial which is not unexpected due to the short length of this subcipher (Table 8).

Table 7 Sub-optimal differential characteristics for 13-round LEA

Rounds	13-round	
—	$\Delta x_0 \| \| \Delta x_1 \| \| \Delta x_2 \| \| \Delta x_3$	$\log_2 p_r$
0	0xC0000000C0400080404000010404000012	—
1	0x80010000800000C40000004C0000000	−13
2	0x02001800820000008000000080010000	−8
3	0x00300100001000000000200002001800	−4
4	0x000200000001FF000040010000300100	−15
5	0x00020000000200000002000000020000	−25
6	**0x00000000000000000000000000020000**	**−5**
7	0x00000000000000000000400000000000	−1
8	0x00000000000002000000080000000000	−2
9	0x00040000000000300000010000000000	−5
10	0x08002000800000080000002000040000	−7
11	0x00401110C40000000000800408002000	−8
12	0x80222188222004008100140000401110	−14
13	0x04491144051900801028 00A180222088	−20
	$\log_2(P_{\text{char}}) = \sum_r \log_2(p_r)$	−127
	$\log_2(P_{\text{diff}}) >$	−115.86

The bold row is the middle row of the cipher, from which the cipher is divided into two parts.

$$
p(x) = \prod_{i=1}^{3} p_i(x) = p_1(x)p_2(x)p_3(x)
$$

$$
\begin{aligned}
&= 6x^{127} + 18x^{128} + 94x^{129} + 337x^{130} \\
&\quad + 1017x^{131} + 3186x^{132} + 8623x^{133} \\
&\quad + 22673x^{134} + 55008x^{135} + 111568x^{136} \\
&\quad + 254616x^{137} + 463615x^{138} + 866416x^{139} \\
&\quad + 1587582x^{140} + 2581241x^{141} + 3974813x^{142} \\
&\quad + 4929537x^{143} + 3847984x^{144}
\end{aligned} \tag{18}
$$

$$
p_3(x) = x^{20} \tag{17}
$$

Therefore, the probability polynomial of the proposed sub-optimal 13-round characteristic is

Finally, the 13-round differential probability is calculated as

$$
p(x)\Big|_{x=\frac{1}{2}} = 2^{-115.8629} \tag{19}
$$

5 Differential analysis of HIGHT block cipher using MILP method

Hong *et al.* [11] proposed a new block cipher HIGHT with 64-bit block length and 128-bit key length, which is suitable for low-cost, low-power, and ultra-light implementations. HIGHT is approved by the Korea Information Security Agency and is adopted as an International Standard by ISO/IEC 18033-3 [11]. This made this cipher an attractive target for cryptanalysis [35–39].

Table 8 Differential characteristics for 12-round HIGHT

Rounds	First characteristic [29]		Second characteristic (new)	
—	$\Delta x_0 \| \| \Delta x_1 \| \| \cdots \| \| \Delta x_6 \| \| \Delta x_7$	$\log_2 p$	$\Delta x_0 \| \| \Delta x_1 \| \| \cdots \| \| \Delta x_6 \| \| \Delta x_7$	$\log_2 p$
0	0x00008227213AEE01	—	0xB000C003000081E2	—
1	0x000027A03A460100	−6	0x00E803000000E2B0	−3
2	0x0000A0B84E010000	−6	0xE80700000000D002	−8
3	0x0000B8C801000000	−4	0x0E00000000000279	−4
4	0x0000C80100000000	−4	0x0000000000007907	−1
5	0x0000010000000000	−3	0x0000000000000700	−5
6	**0x0001000000000000**	**−1**	**0x0000000000010000**	**−3**
7	0x0100000000000082	−2	0x0000008201000000	−2
8	0x00000000009C8201	−3	0x009C820100000000	−3
9	0x000000039C7A0100	−8	0x9C7A010000000003	−8
10	0x00E803BC7A010000	−5	0x7A01000000E803BC	−5
11	0xE800BCF801000002	−6	0x01000002E800BCF8	−6
12	0x00B6F80100B002E8	−5	0x009002E800B6F801	−5
	$\log_2(P_{\text{char}}) = \sum_r (p_r)$	−53	$\log_2(P_{\text{char}}) = \sum_r (p_r)$	−53

The bold row is the middle row of the cipher, from which the cipher is divided into two parts.

The bold row is the middle row of the cipher, from which the cipher is divided into two parts.

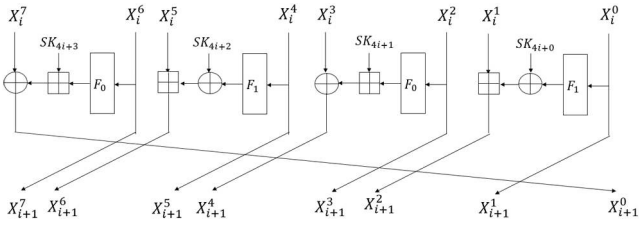

Fig. 4 *Round function of HIGHT cipher*

Table 9 Differential characteristics for 13-round HIGHT

Rounds	First characteristic		Second characteristic	
—	$\Delta x_0 \| \| \Delta x_1 \| \| \cdots \| \| \Delta x_6 \| \| \Delta x_7$	$\log_2 p$	$\Delta x_0 \| \| \Delta x_1 \| \| \cdots \| \| \Delta x_6 \| \| \Delta x_7$	$\log_2 p$
0	0x01004483E20084F2	—	0xEB00805E030084AC	—
1	0x000083E20080F201	−3	0x20005E030000AB3B	−3
2	0x0000E2C1804A0100	−9	0x004A030000003B20	−7
3	0x0000C1184A010000	−6	0x4A010000000020B8	−10
4	0x000018C801000000	−6	0x010000000000B8C8	−4
5	0x0000C80100000000	−4	0x000000000000C801	−4
6	0x0000010000000000	−3	0x0000000000000100	−3
7	**0x0001000000000000**	**−1**	**0x0000000000010000**	**−1**
8	0x0100000000000082	−2	0x0000008201000000	−2
9	0x00000000009C8201	−3	0x009C820100000000	−3
10	0x000000039C7A0100	−8	0x9C7A010000000003	−8
11	0x00E803BC7A010000	−5	0x7A01000000E803BC	−5
12	0xE800BCF801000002	−6	0x01000002E800BCF8	−6
13	0x00B6F80100B002E8	−5	0x009002E800B6F801	−5
	$\log_2(P_{\text{char}}) = \sum_r (p_r)$	−61	$\log_2(P_{\text{char}}) = \sum_r (p_r)$	−61

Although HIGHT has received much attention from the cryptanalysers, a little work has focused on finding the best possible differential characteristic. The first one is the designers' analysis [11], where an evaluation of differential attack is provided by linearising it. According to this analysis, without any discussion about the differential probability, the best differential characteristic found for 11-round of HIGHT has been reported with the probability of 2^{-58}. The other one is a recent one [29] in which differential characteristics are found using a so-called refined MILP model for up to 13 rounds of HIGHT. In the rest of this section, we introduce new 11-round, 12-round, and 13-round differential characteristics/differentials found using our efficient MILP model.

5.1 HIGHT specifications

HIGHT has a 32-round iterative structure, which is a variant of generalised Feistel network. Whitening keys are applied before the first round and after the last round. One round of HIGHT is shown in Fig. 4, where $\left(X_7^i \middle| X_6^i\right|, \ldots, \left|X_0^i\right)$ and $(\mathrm{SK}_{4i+3}|\mathrm{SK}_{4i+2}|\mathrm{SK}_{4i+1}|\mathrm{SK}_{4i})$ indicate the 64-bit input and 32-bit subkey of the ith round, respectively. Each word in HIGHT is a byte. Two subkeys SK_{4i+1} and SK_{4i+3} are added to the data in mod 2^8 while the two other ones are XORed to data. F_0 and F_1 are two linear functions with 8 bits input and 8 bits output which work as follows:

$$\begin{aligned} F_0(x) &= (x <<< 1) \oplus (x <<< 2) \oplus (x <<< 7) \\ F_1(x) &= (x <<< 3) \oplus (x <<< 4) \oplus (x <<< 6) \end{aligned} \tag{20}$$

5.2 MILP-based search for characteristics and differentials of HIGHT

The set of operations used in one round of HIGHT is as follows: two modular additions, two modular additions with one constant input (discussed in Section 2.3), two XORs, two F_0 functions, two F_1 functions, and a final swapping. There are also two XOR operations with subkeys which are effect less in differential attack and would be omitted from our model. Summing up all the constraints related to the above operations, our model has a number of $50n - 32$ constraints for one round where $n = 8$. The reader should be noted that using the more efficient model given in Section 2.3, the amount of reduction in the number of constraints for a cipher can be formulated as follows. Suppose that each round of the cipher contains k modular additions with a constant input, then the amount of reduction in the number of constraints for r rounds of the cipher would be $kr(8(n-1))$. In case of HIGHT, where $k = 2$ and $n = 8$, it would be $112r$.

Similar to LEA, it is impossible to solve an 11-round MILP model as a whole by our machine. So, again searching for the sub-optimal solutions explained in Section 3.2 would be a reasonable strategy here. In the following, our results on 11, 12, and 13 rounds of HIGHT are reported.

5.2.1 Analysis of 11- and 12-round HIGHT: According to the rule of sub-optimal solution searching, we divide the 12-round cipher into two six-round subciphers and independently search each of them. The best 12-round differential characteristics have probability 2^{-53} and there are two such characteristics for HIGHT, one of which was found in [29]. In the first one, reported in [29] too, the internal difference at round six is $\Delta_6^* = (0\text{x}0001, 0\text{x}0000, 0\text{x}0000, 0\text{x}0000)$ and in the other one, reported for the first time in this study, it is $\Delta_6^* = (0\text{x}0000, 0\text{x}0000, 0\text{x}0001, 0\text{x}0000)$. These two characteristics have been shown in Table 9.

Having found the sub-optimum characteristics, we run Algorithm 1 for this model to compute the differential probability. For the second case, the probability polynomials of subciphers are derived as follows:

$$\begin{aligned} p_1(x) &= x^{24} + x^{33} + x^{35} + x^{38} + x^{39} + 3x^{41} + 2x^{42} \\ p_2(x) &= x^{29} + 4x^{40} + 4x^{41} + 4x^{42} + 15x^{43} \end{aligned} \tag{21}$$

and the probability polynomial of the 12-round characteristic is

$$
\begin{aligned}
p(x) = p_1(x)p_2(x) = {} & x^{53} + x^{62} + 5x^{64} + 4x^{65} + 4x^{66} + 16x^{67} \\
& + x^{68} + 3x^{70} + 2x^{71} + 4x^{73} + 4x^{74} + 8x^{75} \\
& + 4x^{76} + 19x^{77} + 19x^{78} + 8x^{79} + 8x^{80} \\
& + 31x^{81} + 35x^{82} + 20x^{83} + 53x^{84} + 30x^{85}
\end{aligned} \tag{22}
$$

An interesting observation about HIGHT is that despite LEA, it does not show a strong differential effect. The probability polynomial is a sparse one which means that a small number of characteristics with insignificant probabilities match this differential. Hence, the differential probability is approximately equal to its only dominant characteristic probability which is equal to

$$
p(x)\big|_{x=\frac{1}{2}} \simeq 2^{-53} \tag{23}
$$

To have a comparison with 11-round characteristic found in [11], we can omit the last round of these 12-round characteristics to come up with an 11-round characteristic with probability 2^{-47}. However, we repeated the search for a sub-optimal solution for the 11-round problem and found a characteristic with probability 2^{-45}, which is much more efficient than that found in [11] with probability 2^{-58}.

5.2.2 Analysis of 13-round HIGHT: The 13-round sub-optimal characteristic of HIGHT would be found by dividing it into 7- and 6-round subciphers, respectively. These characteristics have probability 2^{-61} and are reflected in Table 9. The best two characteristics constrained to have weight one in the middle (round 7). For the 13-round case, the best found characteristics have the differences $\Delta_7^* = (0x0000, 0x0000, 0x0001, 0x0000)$ or $\Delta_7^* = (0x0001, 0x0000, 0x0000, 0x0000)$ at round 7, again. Furthermore, their propagation patterns in the second subcipher, are exactly the same as the 12-round characteristics. However, their 7-round upward patterns are completely different. It means that the sub-optimum 13-round characteristic is not necessarily obtained by extending the sub-optimum 12-round characteristic for one round. Although the probability of these proposed 13-round characteristics are the same as in [29], both the characteristics are new. Now, we compute the probability polynomials of the two subciphers as follows:

$$
\begin{aligned}
p_1(x) &= x^{32} + x^{38} + x^{43} + 2x^{44} \\
p_2(x) &= x^{29} + 4x^{40} + 4x^{41} + 4x^{42} + 15x^{43}
\end{aligned} \tag{24}
$$

Also, the probability of differential as

$$
p(x)\big|_{x=\frac{1}{2}} = p_1(x)p_2(x)\big|_{x=\frac{1}{2}} = \simeq 2^{-61} \tag{25}
$$

The above information is related to the second 13-round characteristic.

5.2.3 Discussion on the solution time: The time taken to solve a MILP, apart from the number of variables and constraints, depends on many other factors such as amount of sparsity and symmetry of the constraints, the gap between the associated LP bound and the optimum value, and the solving algorithm which is used by the solver.

Therefore, to evaluate how our proposed MILP formulation in Section 2.3 affects the speed of solving the problem, we modelled the problem of cryptanalysis of r-round HIGHT in two scenarios, the former is the original one with $480r$ constraints and the second one is the proposed one with $368r$ constraints. We observed that the solution time of our proposed model is, on average, half of the initial model for the same problem.

6 Conclusion

This work gave a more precise analysis of the differential property of ARX ciphers using the MILP technique. We improved the general MILP model for modular addition in a special case and came up with a simpler and faster solvable model. Two block ciphers LEA and HIGHT were studied as instances of ARX ciphers, the results of which were improved or extended.

The MILP model constructed to find the (sub-)optimal characteristic is intrinsically utilisable for computing the differential probability, as well. We enjoyed this capability and investigated the differential effect in these two ciphers. Our findings show that despite LEA which has a strong differential effect, HIGHT does not show such an effect.

7 References

[1] Aumasson, J.P., Henzen, L., Meier, W., *et al.*: 'SHA-3 proposal blake', Submission to NIST, 2008

[2] Ferguson, N., Lucks, S., Schneier, B., *et al.*: 'The skein hash function family'. Submission to NIST (round 3), 2010, 7, (7.5), p. 3

[3] Bernstein, D.J.: 'The Salsa20 family of stream ciphers', in '*New stream cipher designs*' (Springer, Berlin, Heidelberg, 2008), pp. 84–97

[4] Beaulieu, R., Treatman-Clark, S., Shors, D., *et al.*: 'The SIMON and SPECK lightweight block ciphers'. 2015 52nd ACM/EDAC/IEEE Design Automation Conf. (DAC), San Francisco, CA, USA., June 2015, pp. 1–6

[5] Aumasson, J.P., Bernstein, D.J.: 'SipHash: a fast short-input PRF'. Int. Conf. on Cryptology in India, Kolkata, India, December 2012, pp. 489–508

[6] Mouha, N., Mennink, B., Van Herrewege, A., *et al.*: 'Chaskey: an efficient MAC algorithm for 32-bit microcontrollers'. Int. Workshop on Selected Areas in Cryptography, Montreal, QC, Canada, August 2014, pp. 306–323

[7] Bernstein, D.J.: 'ChaCha, a variant of Salsa20'. Workshop Record of SASC, Lausanne, Switzerland, January 2008, Vol. 8, pp. 3–5

[8] Wu, H.: 'The stream cipher HC-128' in Robshaw, M., Billet, O. (Eds.): '*New stream cipher designs*' (Springer, Berlin, Heidelberg, 2008), pp. 39–47

[9] Hong, D., Lee, J.K., Kim, D.C., *et al.*: 'LEA: A 128-bit block cipher for fast encryption on common processors'. Int. Workshop on Information Security Applications, Jeju Island, Republic of Korea, August 2013, pp. 3–27

[10] Rivest, R.L.: 'The RC5 encryption algorithm'. Int. Workshop on Fast Software Encryption, Leuven, Belgium, December 1994, pp. 86–96

[11] Hong, D., Sung, J., Hong, S., *et al.*: 'HIGHT: a new block cipher suitable for low-resource device'. Int. Workshop on Cryptographic Hardware and Embedded Systems, Yokohama, Japan, October 2006, pp. 46–59

[12] Matsui, M.: 'On correlation between the order of S-boxes and the strength of DES'. Workshop on the Theory and Application of Cryptographic Techniques, Perugia, Italy, May 1994, pp. 366–375

[13] Biryukov, A., Nikolić, I.: 'Search for related-key differential characteristics in DES-like ciphers'. Int. Workshop on Fast Software Encryption, Lyngby, Denmark, February 2011, pp. 18–34

[14] Mouha, N., Preneel, B.: 'Towards finding optimal differential characteristics for ARX: Application to Salsa20', Cryptology ePrint Archive, Report 2013/328, 2013

[15] Aumasson, J.P., Jovanovic, P., Neves, S.: 'Analysis of NORX: investigating differential and rotational properties'. Int. Conf. on Cryptology and Information Security in Latin America, Florianópolis, Brazil, September 2014, pp. 306–324

[16] Kölbl, S., Leander, G., Tiessen, T.: 'Observations on the SIMON block cipher family'. Annual Cryptology Conf., Santa Barbara, CA, USA., August 2015, pp. 161–185

[17] Fu, K., Wang, M., Guo, Y., *et al.*: 'MILP-based automatic search algorithms for differential and linear trails for speck'. Int. Conf. on Fast Software Encryption, Bochum, Germany, March 2016, pp. 268–288

[18] Cui, T., Jia, K., Fu, K., *et al.*: 'New automatic search tool for impossible differentials and zero-correlation linear approximations'. IACR Cryptology ePrint Archive, 2016, **2016**, p. 689

[19] Mouha, N., Wang, Q., Gu, D., *et al.*: 'Differential and linear cryptanalysis using mixed-integer linear programming'. Int. Conf. on Information Security and Cryptology, Beijing, People's Republic of China, November 2011, pp. 57–76

[20] Wu, S., Wang, M.: 'Security evaluation against differential cryptanalysis for block cipher structures'. IACR Cryptology ePrint Archive, 2011, p. 551

[21] Sun, S., Hu, L., Wang, P., *et al.*: 'Automatic security evaluation and (related-key) differential characteristic search: application to SIMON, PRESENT, LBlock, DES (L) and other bit-oriented block ciphers'. Int. Conf. on the Theory and Application of Cryptology and Information Security, Kaoshiung, Taiwan, R.O.C., December 2014, pp. 158–178

[22] Sun, S., Hu, L., Wang, M., *et al.*: '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**, p. 2014

[23] Sun, S., Hu, L., Song, L., *et al.*: 'Automatic security evaluation of block ciphers with S-bP structures against related-key differential attacks'. Int. Conf. on Information Security and Cryptology, Guangzhou, People's Republic of China, November 2013, pp. 39–51

[24] Sasaki, Y., Todo, Y.: 'Tight bounds of differentially and linearly active S-boxes and division property of Lilliput', *IEEE Trans. Comput.*, 2018, **67**, (5), pp. 717–732

[25] Xiang, Z., Zhang, W., Bao, Z., *et al.*: 'Applying MILP method to searching integral distinguishers based on division property for 6 lightweight block ciphers'. Int. Conf. on the Theory and Application of Cryptology and Information Security, Hanoi, Vietnam, December 2016, pp. 648–678

[26] Gleixner, A., Eifler, L., Gally, T., *et al.*: 'The SCIP optimization suite 5.0', 2017
[27] Wu, S., Wu, H., Huang, T., *et al.*: 'Leaked-state-forgery attack against the authenticated encryption algorithm ALE'. Int. Conf. on the Theory and Application of Cryptology and Information Security, Bengaluru, India, December 2013, pp. 377–404
[28] Lipmaa, H., Moriai, S.: 'Efficient algorithms for computing differential properties of addition'. Int. Workshop on Fast Software Encryption, Yokohama, Japan, April 2001, pp. 336–350
[29] Yin, J., Ma, C., Lyu, L., *et al.*: 'Improved cryptanalysis of an ISO standard lightweight block cipher with refined MILP modelling'. Int. Conf. on Information Security and Cryptology, Xi'an, China, November 2017, pp. 404–426
[30] Song, L., Huang, Z., Yang, Q.: 'Automatic differential analysis of ARX block ciphers with application to SPECK and LEA'. Australasian Conf. on Information Security and Privacy, Melbourne, VIC, Australia, July 2016, pp. 379–394
[31] Winnen, L.: Sage S-box MILP toolkit
[32] Lai, X., Massey, J.L., Murphy, S.: 'Markov ciphers and differential cryptanalysis'. Workshop on the Theory and Application of Cryptographic Techniques, Brighton, UK., April 1991, pp. 17–38
[33] Sun, L., Wang, W., Wang, M.: 'Automatic search of bit-based division property for ARX ciphers and word-based division property'. Int. Conf. on the Theory and Application of Cryptology and Information Security, Hong Kong, People's Republic of China, December 2017, pp. 128–157
[34] Dwivedi, A.D., Srivastava, G.: 'Differential Cryptanalysis in ARX Ciphers, Applications to LEA'. Cryptology ePrint Archive, Report 2018/898.
[35] Zhang, P., Sun, B., Li, C.: 'Saturation attack on the block cipher HIGHT'. Int. Conf. on Cryptology and Network Security, Kanazawa, Japan, December 2009, pp. 76–86
[36] Wen, L., Wang, M., Bogdanov, A., *et al.*: 'Multidimensional zero-correlation attacks on lightweight block cipher HIGHT: improved cryptanalysis of an ISO standard', *Inf. Process. Lett.*, 2014, **114**, (6), pp. 322–330
[37] Chen, J., Wang, M., Preneel, B.: 'Impossible differential cryptanalysis of the lightweight block ciphers TEA, XTEA and HIGHT'. Int. Conf. on Cryptology in Africa, Ifrance, Morocco, July 2012, pp. 117–137
[38] Ahmadi, S., Ahmadian, Z., Mohajeri, J., *et al.*: 'Low data complexity biclique cryptanalysis of block ciphers with application to piccolo and HIGHT', *IEEE Trans. Inf. Forensics Secur.*, 2014, **9**, (10), pp. 1641–1652
[39] Azimi, S.A., Ahmadi, S., Ahmadian, Z., *et al.*: 'Improved impossible differential and biclique cryptanalysis of HIGHT', *Int. J. Commun. Syst.*, 2018, **31**, (1), p. e3382