IET Information Security

Research Article

IET Journals
The Institution of Engineering and Technology

Division cryptanalysis of block ciphers with a binary diffusion layer

ISSN 1751-8709
Received on 8th January 2017
Revised 21st April 2018
Accepted on 24th July 2018
E-First on 2nd November 2018
doi: 10.1049/iet-ifs.2018.5151
www.ietdl.org

Wenying Zhang[1] ✉, *Vincent Rijmen*[2]

[1]*School of Information Science and Engineering, Shandong Normal University, Jinan, People's Republic of China*
[2]*Department Electrical Engineering (ESAT), KU Leuven and Imec, Leuven, Belgium*
✉ *E-mail: wenyingzh@sohu.com*

Abstract: In this study, the authors propose an accurate approach to model the propagation of the division property of linear layers by the smallest amount of inequalities. The solutions of the inequalities are exactly the division trails of a linear transformation. Therefore, the description is compact and optimal. As applications of their results, they present a 7-round integral distinguisher for both Midori64 and Midori128. The designers of Midori only obtained a 3.5-round integral characteristic. For Skinny64, they find a 10-round integral distinguisher which was previously found by the designers. It is well to remind that their result proves that 7 rounds and 10 rounds are the upper bounds of Midori and Skinny64 correspondingly when searching for integral distinguishers based on division property. The significance of their result lies in that they shed light on how far division cryptanalysis can influence the security analysis of block ciphers with a binary diffusion layer, and their technique can be used to prove security against division cryptanalysis.

1 Introduction

Recently, in order to reduce the energy consumed in data processing, block cipher designers have started to use binary matrices on finite fields as the diffusion layer. The most typical examples are Midori [1], proposed at ASIACRYPT 2015 and Skinny [2], proposed at CRYPTO 2016. The main goal of Skinny is to offer an alternative to the National Security Agency (NSA) design SIMON in terms of hardware/software performance [3]. With their reputation for reaching the requirements of low latency as well as fast diffusion [2], the diffusion layer of SKINNY is an important highlight of its design strategy. Therefore, it is of great importance to evaluate the resistance of ciphers using binary matrices to known cryptanalysis and to give a proof of their security.

The division property [4] is a generalised integral property initially proposed by Todo at EUROCRYPT 2015. At FSE 2016, Todo and Morii proposed the bit-based division property and applied it to find a 14-round integral distinguisher for SIMON32 [5]. At CRYPTO 2016, Boura and Canteaut [6] came up with a new approach by introducing the notion of parity sets, permitting the formulation and characterisation of the division property of any order in a simple way, especially for the construction of the division trails of S-boxes. At ASIACRYPT 2016, Xiang *et al.* [7] proposed a method to characterise the *bit-based* division property with the mixed integer linear programming (MILP) model, which successfully overcomes the difficulty of huge time and memory complexities of utilising the bit-based division property in a security evaluation. They accurately described the division-property propagations by choosing an appropriate objective function and analysed six block ciphers with bit-permutation diffusion layers. They left the feasibility of MILP method applied to ciphers with diffusion layers that are not bit permutations as a future work. Soon after, Sun *et al.* [8, 9] handled the feasibility of MILP-aided division property for primitives with non-bit-permutation linear layers. They successfully extended the MILP method to exclusive-OR (XOR)-based and Addition-Rotation-XOR operation (ARX)-based structures by introducing some intermediate variables in the linear layer, building $2n$ inequalities for the n-bit linear layers.

However, we found that the solutions of linear inequalities in [8] contain some impossible division trails, which may eventually lead the search for integral trails to a premature end and result in a shorter integral distinguisher. In the following, for block ciphers with a binary linear layer, we will give a compact description [10] of the propagation of the division property through the linear layer. Combined with our compact description, the method in [8, 9] will be more precise and closer to perfection.

1.1 Our contributions

i. We model the propagation of the division property through a linear layer by constructing linear inequalities from the XOR operations described by the matrix of the linear layer, so that their solutions exactly represent all division trails of the linear layer. Just like for an S-box, the solutions set of the linear inequalities is equal to the set of division trails of the linear transformation when taking the linear transformation as a big S-box. We find that there is a one-to-one map from the vectors in the division trails of a linear transformation to invertible submatrices of the matrix $\boldsymbol{M}$, and we can give a simple description of the invertibility of the submatrices by some inequalities.
ii. As applications of our methodology, we propose compact representations for the division properties through the linear layer of Midori and Skinny. We describe exactly all division trails of their linear layers without any extra parasitical one. As for Midori64 and Midori128, the designers have obtained a 3.5-round integral characteristic. In comparison with the former method, we find 7-round integral characteristics, twice what the former method achieves, for both Midori64 and Midori128. In Eurocrypt 2017, Sasaki and Todo [11] found a 7-round truncated impossible differential for Midori128 by an automated tool, but there is no corresponding result for Midori64. For Skinny64, we find the same 10-round integral distinguisher for Skinny64 that the designers have found. The designers of Skinny mentioned in their security analysis that the division property can probably be used to slightly extend their results. Our results, however, show that any improvement in the length of the integral distinguisher using the division property is impossible. A summary of the comparisons of our results with the previous integral characteristics is shown in Table 1.

© The Institution of Engineering and Technology 2018

iii. The highlight of our work is that we found a way to build distinguishers for the linear layer of a block cipher and gave a compact description. Thanks to the compact representation, we can accurately evaluate the propagation of the division property through binary linear layers. Therefore, we proposed a method to check the security of block ciphers against integral cryptanalysis using the division property.

This paper is organised as follows. In Section 2, we give some preliminaries for division property and MILP, introduce Midori and Skinny and model S-boxes in the two ciphers which will be used later. In Section 3, we propose a theoretical compact description and a practical compact description of the division properties of binary linear layers. In Section 4, we apply the method in Section 3 of the Midori family of block ciphers and show the improvements. In Section 5, we present a 10-round integral distinguisher for Skinny64. In Section 6, we discuss the application of our method on complicated matrices and propose the technical improvements of our method prior over the preceding ones. Finally, in Section 7, we end with a conclusion and possible future work.

2 Preliminaries and preparations

2.1 Notations and definitions

We present our notations in Table 2.

If $\boldsymbol{u} = (u_1, \ldots, u_n)$ is a vector of $\mathbb{F}_2^n$, we denote by x^u the bit product

$$x^u = \prod_{i=1}^{n} x_i^{u_i}.$$

The division property is defined for a multi-set X and is calculated by summing the bit product function over all vectors of X.

Definition 1: (Division property): A multi-set $X \subseteq \mathbb{F}_2^n$ is said to fulfil the division property [4, 6] of order t, D_t^n for some $1 \leq t \leq n$, if the sum over all vectors x in X of the product x^u equals 0, for all vectors $\boldsymbol{u}$ that have a hamming weight less than t, i.e.

$$\bigoplus_{x \in X} x^u = 0 \text{ for all } u \in F_2^n \text{ such that } \mathrm{wt}(\boldsymbol{u}) < t.$$

Notations: In this paper, we consider the case where the inputs of a transformation (for instance: a round function, an S-box and a linear transformation) run over a subspace or an affine subspace of $\mathbb{F}_2^n$. In particular, the i_1th, …, i_tth components of them each take the values 0 and 1, while the other components are fixed. We denote the division property by D_k^n, where $\boldsymbol{k} \in \mathbb{F}_2^n$, $wt(\boldsymbol{k}) = t$ and the i_1th, …, i_tth components of $\boldsymbol{k}$ are 1 and the others are 0. For example, $D_{(0101)}^4$ represents the inputs of a transformation on $\mathbb{F}_2^4$ are formed as for (x_1, c_2, x_3, c_4), where x_1, x_3 vary from 0 to 1, c_2, c_4 is fixed.

Definition 2: (Division Trail): Let f_r be the round function of an iterated block cipher. Let $\mathbb{K}_i$, $0 \leqslant i \leqslant r$ be a set of vectors in $\mathbb{Z}^m$. Assume the input multi-set to the block cipher has initial division property D_k^n, and denote the division property after i rounds of propagation through f_r by $D_{\mathbb{K}_i}^n$. Thus, we have the following chain of division-property propagations:

$$\{\boldsymbol{k}\} \triangleq \mathbb{K}_0 \xrightarrow{f_r} \mathbb{K}_1 \xrightarrow{f_r} \mathbb{K}_2 \xrightarrow{f_r} \cdots \xrightarrow{f_r} \mathbb{K}_r.$$

Moreover, for any vector $\boldsymbol{k}_i^*$ in $\mathbb{K}_i (i \geq 1)$, there must exist a vector $\boldsymbol{k}_{i-1}^*$ in $\mathbb{K}_{i-1}$ such that $\boldsymbol{k}_{i-1}^*$ can propagate to $\boldsymbol{k}_i^*$ by division property propagation rules. Furthermore, for $(\boldsymbol{k}_0, \boldsymbol{k}_1, \ldots, \boldsymbol{k}_r) \in \mathbb{K}_0 \times \mathbb{K}_1 \times \cdots \times \mathbb{K}_r$, if $\boldsymbol{k}_{i-1}$ can propagate to $\boldsymbol{k}_i$ for all $i \in \{1, 2, \ldots, r\}$, we call $(\boldsymbol{k}_0, \boldsymbol{k}_1, \ldots, \boldsymbol{k}_r)$ an r-round division trail. Moreover, $\mathbb{K}_0$ is called the initial division property [4, 7, 8].

Table 1 Our results on Skinny and Midori, and compared with previous results

Cipher	Rounds (previous)	Rounds (ours)
Midori64	3.5 [1]	7 (=upper bound)
Midori128	3.5 [1]	7 (=upper bound)
Skinny64	10 [2]	10 (=upper bound)

Table 2 Notations used throughout this paper

$\boldsymbol{x} = (x_1, \ldots, x_n)$	an n-bit boolean vector
$\mathrm{wt}(\boldsymbol{x})$	Hamming weight of the boolean vector x
$\boldsymbol{M}$	matrix
$\boldsymbol{x}^{\mathrm{T}}$	transposition of x
$\hat{\boldsymbol{M}}$	submatrix of $\boldsymbol{M}$
e_i	unit vector whose the ith bit is 1
$\mathrm{wt}(r_i)$	Hamming weight of the ith row of $\boldsymbol{M}$
$\mathbb{R}$	set of real numbers
$\mathbb{F}_2^n$	set of all n-bit boolean vectors
$\mathbb{F}(2^n)$	finite field of size 2^n
$\mathbb{N}$	set of natural numbers
$\mathbb{Z}$	set of integer numbers
t	integer
$\boldsymbol{k}$	boolean vector

Division trails are vectors which show the propagation path of the division property in the process of encryption, showing the balancedness of intermediate states.

The propagation of the division property through a round function of the block cipher is actually a series of transitions of vectors. Assume that the input multi-set to the S-box has division property $D_{k_0}^n$, and the output multi-set has division property $D_{k_1}^n$, where $\boldsymbol{k}_0 = (x_1, \ldots, x_n)$, $\boldsymbol{k}_1 = (y_1, \ldots, y_n)$, then we call $(x_1, \ldots, x_n, y_1, \ldots, y_n)$ a division trail of this S-box [7, 10]. Especially, when the S-box is a linear transformation, we call $(x_1, \ldots, x_n, y_1, \ldots, y_n)$ a division trail of the linear transformation. For the details of the construction of division trails, please see [6, Section 5.3] or [10, Section 2.4].

2.2 MILP and integral cryptanalysis

Now, we briefly introduce MILP, MILP is a general mathematical tool, which takes an objective function and a system of linear inequalities with respect to real numbers as input, and searches for an optimal solution which minimises/maximises the objective function satisfying all the inequalities. Mouha *et al.* [12] showed that the problem of finding the optimal differential path can be converted to MILP. An MILP problem can formally be described as follows.

MILP: Find a vector $\boldsymbol{x} \in \mathbb{Z}^k \times \mathbb{R}^{n-k} \subseteq \mathbb{R}^n$ with $Ax \leq b$, so that the linear function

$$c_1 x_1 + c_2 x_2 + \cdots + c_n x_n$$

is minimised (or maximised), where $(c_1, \ldots, c_n) \in \mathbb{R}^n$, $A \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$.

In recent years, MILP has been explicitly applied in varieties of cryptographic research areas [11–13]. We are mainly concerned about the application of the MILP method in integral cryptanalysis. Integral cryptanalysis was described by Knudsen and Wagner in [14]. There are two major techniques to construct an integral characteristic; one uses the propagation characteristic of integral properties [14, 15] and the other estimates the algebraic degree [16]. In this paper, we study the propagation of the integral property.

© The Institution of Engineering and Technology 2018

Table 3 Specifications of S_4, Sb_0 and Sb_1

x	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
$S_4(x)$	c	6	9	0	1	a	2	b	3	8	5	d	4	e	7	f
$Sb_0(x)$	c	a	d	3	e	b	f	7	8	9	1	5	0	2	4	6
$Sb_1(x)$	1	0	5	3	e	2	f	7	d	a	9	b	c	8	4	6

In the MILP progress, we have to give an initial division property D_k^n and a stopping rule by constructing an objective function in terms of the Hamming weight of the division trail.

Stopping rule. Let $D_{k_i}^n$ be the output division property after i rounds of encryption. Let $D_{k_0}^n$ be the input division property of the first round. If $\boldsymbol{k}_{r+1}$ *contains all the n unit vectors for the first time*, it means that none of the n bits of output is balanced, and the division-property propagation should stop and an r-round distinguisher can be derived from $D_{k_r}^n$.

2.3 Midori block cipher

Midori is a family of lightweight block ciphers recently published at ASIACRYPT 2015. They follow the Substitution-permutation network (SPN) structure and have been advertised as one of the first lightweight ciphers optimised with respect to the energy consumed by the circuit per bit in the encryption or decryption operation. To achieve the desired low-energy goal, several design decisions were made such as using a diffusion layer consisting of almost-Maximum Distance Separable (MDS) 4×4 binary matrices. The Midori family consists of two ciphers: Midori64 and Midori128. The block sizes are 64 and 128 bits and the number of rounds is 16 and 20, respectively, and the key size is 128 bits for both. The plaintext and the intermediate state are described by matrices of size 4×4

$$\begin{pmatrix} s_0 & s_4 & s_8 & s_{12} \\ s_1 & s_5 & s_9 & s_{13} \\ s_2 & s_6 & s_{10} & s_{14} \\ s_3 & s_7 & s_{11} & s_{15} \end{pmatrix}.$$

The size of each cell is 4 bits for Midori64 and 8 bits for Midori128.

The round function consists of the four operations SubCell, ShuffleCell, MixColumn and KeyAdd that update the n-bit state S:

i. *SubCell*: Apply the 4 bit S-box Sb_0 and 8 bit S-box $SSb_{i(\mathrm{mod}4)}$ to each cell of Midori64 and Midori128, respectively. The truth tables of Sb_0 and Sb_1 are listed in Table 3.
ii. *ShuffleCell*: Each cell of the state is shuffled as follows: $(s_0, s_1, \ldots, s_{15}) \leftarrow (s_0, s_{10}, s_5, s_{15}, s_{14}, s_4, s_{11}, s_1, s_9, s_3, s_{12}, s_6, s_7, s_{13}, s_2, s_8)$.
iii. *MixColumn*: Multiply each column by a 4×4 matrix $\boldsymbol{M}_{\mathrm{Midori}}$ over the finite field $\mathbb{F}(2^4)$ and $\mathbb{F}(2^8)$ correspondingly, where $\boldsymbol{M}_{\mathrm{Midori}}$ is

$$\boldsymbol{M}_{\mathrm{Midori}} = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}.$$

2.4 Skinny block cipher

Skinny [2] is a family of lightweight block ciphers proposed at Crypto 2016. It adopts the SPN structure just like Advanced Encryption Standard (AES). Skinny has variable block sizes of 64 and 128 bits, and key sizes of 64, 192 or 256 bits. In this paper, for simplicity, we just focus on the 64 bit block size. The 64 bit plaintext and the intermediate state are described by nibble matrices of size 4×4

$$\begin{pmatrix} m_0 & m_1 & m_2 & m_3 \\ m_4 & m_5 & m_6 & m_7 \\ m_8 & m_9 & m_{10} & m_{11} \\ m_{12} & m_{13} & m_{14} & m_{15} \end{pmatrix}.$$

Each round of Skinny is composed of four operations applied to the internal state in the order specified below:

i. *SubByte*: Apply the 4 bit S-box S_4 (Table 3) to each nibble.
ii. *AddConstants and AddRoundKey(AK)*: XOR the state with constant and subkey.
iii. *ShiftRow*: Shift the ith row by i nibbles to the right $i = 0, 1, 2, 3$.
iv. *MixColumn*: Multiply each column by a constant 4×4 matrix $\boldsymbol{M}_{\mathrm{Skinny}}$ over the field $\mathbb{F}(2^4)$, where

$$\boldsymbol{M}_{\mathrm{Skinny}} = \begin{pmatrix} 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \end{pmatrix}.$$

For completeness, we list the primitive representation of $\boldsymbol{M}_{\mathrm{Skinny}}$ in Section 12, by 'primitive representation' we mean the original $\boldsymbol{M}_{\mathrm{Skinny}}$, viewed as a matrix with entries from $\mathbb{F}(2)$. Since XORing with constants does not influence the division property, we do not consider AddConstants (AC) and AddRoundKey (AK) in our analysis. For more details about Midori and Skinny, please see [1, 2], respectively.

2.5 Modelling S-boxes

To use MILP in block cipher evaluation, a critical step is to build a set of linear inequalities $Ax \leq b$ to describe division property propagations for the round function. A compact representation of the round function should satisfy the following two conditions according to Xiang *et al.* [7].

C1: Each division trail must satisfy all linear inequalities of the linear inequality system. That is, each division trail corresponds to a solution of the linear inequalities.
C2: Each solution of the linear inequalities corresponds to a division trail. That is the set of all solutions of the linear inequalities which does not contain any impossible division trail.

Since XORing with constants and subkeys does not influence the propagation of division property, and ShuffleCell just shuffles the division trails, we just need to model the S-box and the diffusion layer.

For the propagation through an S-box, we apply the table-aided bit-based division property introduced in [6] to generate the propagation table of the S-box. After that, just as what has been introduced in [7], by using the *inequality_generator()* function in the Sage software, a set of linear inequalities is returned. Furthermore, this set can be reduced by the greedy algorithm (Algorithm 1) in [13]. The inequalities $\mathfrak{L}_1$ are the five inequalities used to describe the Midori64 S-box $Sb_0(x)$. Their solutions are exactly the 48 division trails of $Sb_0(x)$. The inequalities $\mathfrak{L}_2$ are the ten inequalities used to describe the Midori128 S-box $Sb_1(x)$. Their solutions are exactly the 49 division trails of $Sb(x)$. Since $SSb_i(x)$, $i = 0, 1, 2, 3$ are processed by two $Sb_1(x)$ in parallel, we need 20 inequalities for each 8 bit S-box used in Midori128. The inequalities $\mathfrak{L}_3$ in Section 9 are the 12 inequalities used to describe

© The Institution of Engineering and Technology 2018

the Skinny S-box. Their solutions are exactly the 44 division trails of S_4.

3 Compact characterisation of division trails through binary linear layers

In block ciphers with bit-based permutations, the division property of bit i propagates to the bit $P(i)$ when the bit i is moved to the bit $P(i)$. Therefore, modelling the S-box is enough for the modelling of the round function. However, for more general linear diffusion layers, modelling the linear layer is a distinctive and sophisticated step of MILP-based cryptanalysis.

The most popular idea to describe the division property of a linear layer is to introduce some intermediate binary variables $t_{k_{(i,\,j)}}$, $1 \le k_{(i,\,j)} \le \mathrm{wt}(M)$. For example, the equations used in [8] are

$$\begin{cases} y_i = \sum\limits_{a_{i,\,*} \neq 0} t_{k_{(i,\,*)}}, & 1 \le i \le n \\ x_j = \sum\limits_{a_{*,\,j} \neq 0} t_{k_{(*,\,j)}}, & 1 \le j \le n \\ y_i,\ x_j,\ t_{k_{i,\,j}} \text{ are binaries}. \end{cases} \tag{1}$$

However, the following example will show that (1) does not include enough inequalities.

Example 1: We take the following 4 bit linear transformation as an example: $L\colon (x_1, x_2, x_3, x_4) \mapsto (x_3 + x_1,\ x_3 + x_2 + x_1,\ x_3 + x_2,\ x_4 + x_3)$. It is an S-box with algebraic degree 1:

$$\begin{pmatrix} y_4 \\ y_3 \\ y_2 \\ y_1 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_4 \\ x_3 \\ x_2 \\ x_1 \end{pmatrix}. \tag{2}$$

According to (1), one can introduce intermediate variables t_i, $1 \le i \le 9$ and get the following equations in binary variables:

$$\begin{cases} y_4 = t_1 + t_2 \\ y_3 = t_3 + t_6 \\ y_2 = t_4 + t_7 + t_8 \\ y_1 = t_5 + t_9 \\ x_4 = t_1 \\ x_3 = t_2 + t_3 + t_4 + t_5 \\ x_2 = t_6 + t_7 \\ x_1 = t_8 + t_9 \\ x_i,\ y_j,\ t_{k_{(i,\,j)}} \text{ are binaries}. \end{cases} \tag{3}$$

Given (x_1, x_2, x_3, x_4), one can get (y_1, y_2, y_3, y_4) by (3). Table 4 lists all the 44 solutions of linear system (3), where vectors on $\mathbb{F}_2^4$ are in hexadecimal notation. However, Table 4 does not show correctly the propagation characteristic of the bit-based division property of the linear transformation L as an S-box.

Indeed, the four vectors shown in bold are superfluous; the correct propagation of division property has all vectors excluding them. It must be noted that, in some time, these four vectors may lead to n unit vectors and force the searching process to end prematurely. Now, we explore how this happens. In fact, the cause of the problem is that the left-upper-order three submatrices of $\boldsymbol{M}$ are non-invertible. The ANF of $y_4y_3y_2$ is $(x_4 + x_3)(x_3 + x_2)(x_3 + x_2 + x_1) = x_4x_3x_2 + x_4x_2x_3 + x_3x_4 + x_2x_4 + x_1x_2x_4. + x_1x_3x_4 + x_1x_2x_3 + x_2x_3 + x_1x_3 + x_3$

Although the monomial $x_4x_3x_2$ appears in this expression, it appears twice and will be completely cancelled out. It follows that $y_4y_3y_2$ should not be included in the division table corresponding to the input $x_4x_3x_2$. That is, $(1110, 1110) \triangleq (E, E)$ is not a division trail.

Table 4 Propagation of the bit-based division property for a linear transformation L

k of input D_k^4	k of output D_k^4	k of input D_k^4	k of output D_k^4
0	{0}	8	{8}
1	{1,2}	9	{9,A,}
2	{2,4}	A	{A,C}
3	{3,5,6}	B	{B,D,E}
4	{1,2,4,8}	C	{9,A,C}
5	{5,6,9,A,**3**}	D	{D,E,**B**}
6	{3,5,**6**,A,C}	E	{D,**E**,B}
7	{7,B,D,E}	F	{F}

The four vectors shown in bold are superfluous; they are not the division trails but the others are.

According to the proceeding methods, the division trail is counted by verifying whether a monomial appears at least once in the calculation progress of the ANF, without taking into account possible cancellation.

Hence in the MILP process by (1), one will get extra vectors that do not correspond to division trails. In other words, a part of the solutions is parasitical. Therefore, (1) does not describe the division property through linear layer compactly. We describe now our method.

3.1 Compact theoretical description

Observation: The inherent character of the division property of a linear transform is the independence of variables. By this, we mean the following. If the input of an invertible n-bit permutation runs from 0 to $2^n - 1$, then the output takes all 2^n values. Hence, the n output bits can be described as n independent variables, since they each take the values 0 and 1 often equally, independent of the value of the other output bits. On the contrary, if the input of an n bit permutation cannot take all values from 0 to $2^n - 1$, then the output bits will not act as n independent variables. However, some of the bits may still be independent of one another. Just like an invertible linear transformation maps a space of dimension $\boldsymbol{k}$ to a space of dimension $\boldsymbol{k}$, it holds that if the input of a linear transformation has the division property of order k, then also the output must have the division property of order $\boldsymbol{k}$. Similarly, we propose the following main theorem about how the independence of input variables propagates to the output variables and how this is related to the division property of order $\boldsymbol{k}$.

Theorem 1: Let $\boldsymbol{M} = (a_{i,\,j})$ *be the* $n \times n$ matrix of an invertible linear transformation L. Let $(x, y) = (x_1, \ldots, x_n, y_1, \ldots, y_n) \in \mathbb{F}_2^n \times \mathbb{F}_2^n$, $I_x = \{i, x_i = 1\} = \{i_1, \ldots, i_{\mathrm{wt}(x)}\}$, $I_y = \{j, y_j = 1\} = \{j_1, \ldots, j_{\mathrm{wt}(y)}\}$. Then, (x, y) is one of the division trails of L if and only if the order $\mathrm{wt}(\boldsymbol{x})$ submatrix whose columns indices are taken from I_x and rows indices are taken from I_y is invertible.

Proof: The ith row of $\boldsymbol{M}$ contains the coefficients of the linear transformation $y_i = \sum_{k=1}^{n} a_{i,\,k}x_k$, which is the ith component of the linear transformation, corresponding to the list of all monomials with degree one in the ANF of $x \mapsto y_i(x)$. Since the linear transformation is invertible, it maps a $\boldsymbol{k}$-dimensional subspace onto a $\boldsymbol{k}$-dimensional subspace. Therefore, if (x, y) is one of the division trails, we have $\mathrm{wt}(\boldsymbol{x}) = \mathrm{wt}(y)$.

'⇒' By [6, Proposition 7], if (x, y) is a division trail of an order n linear transformation, then the monomial $\prod_{i \in I_x} x_i$ appears in the expansion of $\prod_{j \in I_y} y_j$, which equals to

$$(a_{j_1,\,1}x_1 + \cdots + a_{j_1,\,n}x_n)\cdots(a_{j_{\mathrm{wt}(x)},\,1}x_1 + \cdots + a_{j_{\mathrm{wt}(x)},\,n}x_n). \tag{4}$$

In the expanded form of (5), the coefficient of $x_{i_1}\cdots x_{i_{\mathrm{wt}(x)}}$ is a sum of terms which take exactly one coefficient as a factor from each factor of

© The Institution of Engineering and Technology 2018

$$(a_{j_1,\,i_1}x_{i_1} + \cdots + a_{j_1,\,i_{\mathrm{wt}(x)}}x_{i_{\mathrm{wt}(x)}})\cdots$$
$$(a_{j_{\mathrm{wt}(x)},\,i_1}x_{i_1} + \cdots + a_{j_{\mathrm{wt}(x)},\,i_{\mathrm{wt}(x)}}x_{i_{\mathrm{wt}(x)}})\,.$$

This results in

$$\sum_{\pi} a_{m_1,\,i_1}a_{m_2,\,i_2}\cdots a_{m_{\mathrm{wt}(x)},\,i_{\mathrm{wt}(x)}}, \tag{5}$$

where $\pi = (m_1, m_2, \ldots, m_{\mathrm{wt}(x)}) \in \mathbb{N}\times\cdots\times\mathbb{N}$ runs over all permutations of $j_1, \ldots, j_{\mathrm{wt}(x)}$. Equation (5) is exactly the definition of the determinant of the $\mathrm{wt}(x)\times\mathrm{wt}(x)$ binary matrix formed by taking the $i_1, \ldots, i_{\mathrm{wt}(x)}$ columns and $j_1, \ldots, j_{\mathrm{wt}(x)}$ rows from $\boldsymbol{M}$

$$\boldsymbol{B} = \begin{pmatrix} a_{j_1,\,i_1} & a_{j_1,\,i_2} & \cdots & a_{j_1,\,i_{\mathrm{wt}(x)}} \\ a_{j_2,\,i_1} & a_{j_2,\,i_2} & \cdots & a_{j_2,\,i_{\mathrm{wt}(x)}} \\ \vdots & \vdots & \cdots & \vdots \\ a_{j_{\mathrm{wt}(x)},\,i_1} & a_{j_{\mathrm{wt}(x)},\,i_2} & \cdots & a_{j_{\mathrm{wt}(x)},\,i_{\mathrm{wt}(x)}} \end{pmatrix}. \tag{6}$$

Hence, if (x, y) is a division trail, then $\det(B) = 1$.

'$\Leftarrow$' If $\det(B) = 1$, then the $i_1, \ldots, i_{\mathrm{wt}(x)}$ columns of $\boldsymbol{M}$ are linearly independent, that is, $x_{i_1}, \ldots, x_{i_{\mathrm{wt}(x)}}$ are linearly independent. Moreover, similarly $y_{j_1}, \ldots, y_{j_{\mathrm{wt}(x)}}$ are linearly independent. According to the observation before Theorem 1, which means that if the input of a linear transformation has the division property $\boldsymbol{x}$, then the output of this linear transformation has the division property y. Hence (x, y) is a division trail. □

Theorem 1 shows that there is a one-to-one correspondence between the invertible submatrices of $\boldsymbol{M}$ and its division trails. The internal relation between the division trail and the matrix is the invertibility of the submatrix. Therefore, checking whether a vector is a division trail of a linear transformation $\boldsymbol{M}$ is equivalent to check whether the corresponding submatrix of $\boldsymbol{M}$ is invertible.

3.2 Compact practical description

So far, we have given a compact theoretical description for division trails of linear layers of block ciphers. In the following, we will describe a practical method to build a series of inequalities such that their solutions are exactly the division trails of the linear transformation.

According to Theorem 1, in order to get all the division trails, it is sufficient to find all the invertible submatrices of $\boldsymbol{M}$. Now, the key point is to have a way to describe the character of the invertible submatrices. Inspired by the way of representing the division trails of S-box as linear inequalities in [7], we want to find a practical way to describe invertible submatrices by inequalities.

In the following, we consider a $ns\times ns$ matrix $\boldsymbol{M} = (a_{i,\,j})$, $1 \le i, j \le ns$ that is derived by starting from a linear map defined by an $s\times s$ matrix $\boldsymbol{M}_s \in (\mathbb{F}(2^n))^{s\times s}$ and then choosing a basis in $\mathbb{F}(2^n)$ and representing all elements of $\mathbb{F}(2^n)$ as n-bit vectors. We consider only the matrix $\boldsymbol{M}$ derived from a matrix $\boldsymbol{M}_s$ with coefficients $\in \{0, 1\}$. Since each division trail (x, y) satisfies $\mathrm{wt}(\boldsymbol{x}) = \mathrm{wt}(y)$, (x, y) has even Hamming weight. A division trail with Hamming weight $2t$ corresponds to an invertible order t submatrix of $\boldsymbol{M}$. We describe the division trails by the invertible submatrices in ascending order of Hamming weight.

3.2.1 Describing the division trails with hamming weight 2:

First, we describe all the invertible submatrices of order 1, that is, all non-zero entries in $\boldsymbol{M}$. They correspond to division trails, where x, y are both basis vectors. Suppose that $a_{i,\,k_1} = \cdots = a_{i,\,k_{\mathrm{wt}(r_i)}}$ are the ones in the ith row of $\boldsymbol{M}$. *The trails are the solutions of the set of inequalities*

$$\begin{cases} \sum_{k=0}^{ns-1} a_{i,\,k}x_k - y_i \ge 0, \\ y_j = 0,\ j \ne i\,. \end{cases} \tag{7}$$

since by (7), when $y_i = 1$, there are at least one of x_k, $k \in \{k_1, \ldots, k_{\mathrm{wt}(r_i)}\}$ equals to 1. Therefore, we get $\mathrm{wt}(r_i)$ vectors $(e_{k_1}, e_i), \ldots, (e_{k_{\mathrm{wt}(r_i)}}, e_i) \in \mathbb{F}_2^{2ns}$. They are all the division trails with Hamming weight 2.

3.2.2 Describing the trails with hamming weight 4:

Corresponding to the division trails with Hamming weight 4, we have to characterise $\boldsymbol{M}$'s invertible submatrices of order 2. It is well known that a matrix is invertible if and only if its row vectors are linearly independent. So we consider the linear combinations of two rows. We divide the rows of $\boldsymbol{M}$ into n cosets. Let $\Lambda_\sigma = \{\sigma, \sigma + n, \ldots, \sigma + (s-1)n\}$, $0 \le \sigma \le n-1$. We first consider the case $i - j \ne 0 (\mathrm{mod}\ n)$, i.e. i and j are in different cosets. Taking into account the fact that $\boldsymbol{M}_s$ is a binary matrix over the finite field $\mathbb{F}(2^n)$, we see that when $i - j \ne 0 (\mathrm{mod}\ n)$ the ith row and the jth row have no common non-zero entries. The inequalities

$$\sum_{k=0}^{ns-1} a_{ik}x_k \ge y_i, \quad \sum_{k=0}^{ns-1} a_{jk}x_k \ge y_j, \quad y_m = 0,\ m \ne i,\ m \ne j$$

given the order two invertible submatrices containing the ith and the jth rows.

Now, we consider the case where i and j are in the same coset. We study the XOR of the ith row and the jth row, with $i = j (\mathrm{mod}\ n)$. If a submatrix includes these rows, then it must contain at least one column k, where $a_{i,\,k} \oplus a_{j,\,k} = 1$ or else the submatrix will be singular. Therefore, it is enough to use

$$\begin{cases} \sum_{k=0}^{ns-1} (a_{i,\,k} \oplus a_{j,\,k})x_k - y_i - y_j \ge -1, \quad i, j \in \Lambda_\sigma \\ y_m = 0, \quad m - \sigma \ne 0 (\mathrm{mod}\ n)\,. \end{cases} \tag{8}$$

to describe this phenomenon.

For example, for the third and the 11th rows in $\boldsymbol{M}_{\mathrm{Skinny}}$ (see Section 11), we have $y_3 + y_{11} = x_3 + x_7 + x_{15}$. The two rows are the same, except for the 3rd, 7th and 15th components. Hence, if a submatrix includes these two rows, at least one of the 3rd, 7th or 15th column should be taken or else the submatrix will be singular. Therefore, it is enough to use $x_3 + x_7 + x_{15} - y_3 - y_{11} \ge -1$, $y_i = 0$, $i \ne 3$, $i \ne 11$ to describe this phenomenon.

3.2.3 Describing the trails with hamming weight $2t$, $3 \le t < s$:

Similar to the argumentations in Section 3.2.2, after collecting the XOR of t special rows of $\boldsymbol{M}$, we get the following inequalities which describe trails with Hamming weight $2t$:

$$\begin{cases} \sum_{k=0}^{ns-1} (a_{i_1,\,k} \oplus \cdots \oplus a_{i_t,\,k})x_k - y_{i_1} - \cdots - y_{i_t} \ge -(t-1), \\ y_m = 0,\ m - \sigma \ne 0 (\mathrm{mod}\ n),\ i_1, \ldots, i_t \in \Lambda_\sigma\,. \end{cases} \tag{9}$$

For $\boldsymbol{M}_{\mathrm{Skinny}}$, we have $x_4 + x_{12} - y_0 - y_4 - y_8 \ge -2$. This implies that if y_0, y_4, y_8 are all equal to 1, then either x_4 or x_{12} is 1; otherwise, the 0, 4 and 8 rows in the submatrix will be linearly dependent, and the submatrix is singular.

3.2.4 Describing the division trails of hamming weight $2s$:

By collecting the XOR of all the $i, i+n, \ldots, i+(s-1)n$ rows of $\boldsymbol{M}$, and taking into account the compression rule of the XOR operation [7], noting that $x_i, x_{i+n}, \ldots, x_{i+(s-1)n}$ are the only inputs of $y_i, y_{i+n}, \ldots, y_{i+(s-1)n}$ and $y_i, y_{i+n}, \ldots, y_{i+(s-1)n}$ only include $x_i, x_{i+n}, \ldots, x_{i+(s-1)n}$, we get the following equations for some special division trails with Hamming weight $2s$:

© The Institution of Engineering and Technology 2018

$$\begin{cases} x_i \oplus x_{i+n} \oplus \cdots \oplus x_{i+(s-1)n} = y_i \oplus y_{i+n} \oplus \cdots \oplus y_{i+(s-1)n} \\ y_j = 0,\ j - i \neq 0(\bmod\ n)\,. \end{cases} \quad (10)$$

3.3 Necessity and sufficiency of the proceeding inequalities

We claim that the four steps in Section 3.2 give all the division trails for the binary linear layer. This is proven in Theorem 2.

Theorem 2: The four steps in Section 3.2 describe exactly the invertible submatrices of the matrix $\boldsymbol{M}$ of the binary linear layer. The number of inequalities used to describe the division trails of the linear layer is $n \times (2^s - 1)$, where s is the order of $\boldsymbol{M}$ over $\mathbb{F}(2^n)$.

Proof: On the one hand, for an invertible submatrix of any order, we can divide its rows indices into the following cosets: $\{0, n, \ldots, (s-1)n\}$, $\{1, n+1, \ldots, (s-1)n+1\}$, $\ldots$, $\{n-1, .$
$2n-1, \ldots, sn-1\}$
Since the rows in different cosets have no common non-zero entries in the same column, we can take into account the XOR operation of rows in each coset separately. For the y_i's with indices in the same coset, we construct inequalities according to Section 3.2. Hence for a submatrix of any order, we have the corresponding inequalities.

On the other hand, the set of linear inequalities, we constructed, forms the optimum selection. By this we mean that it has the smallest number of inequalities; each of these inequalities and each combination of these inequalities stands for a large class of invertible submatrices. If one of them is removed, then some singular submatrices will come in, just like the four bold vectors in Table 4.

Hence, the four steps in Sections 3.2.1–3.2.4 are sufficient and necessary to describe all the invertible submatrices of the linear layer matrix. The number of inequalities for the binary linear layer is

$$n \times \left(\binom{s}{1} + \binom{s}{2} + \cdots + \binom{s}{s} \right) = n \times (2^s - 1)\,.$$

□

4 Application to Midori

In this section, we show the application of our technique on the cryptanalysis of Midori.

4.1 Modelling the linear layer

Each division trail of the linear transformation of Midori64 can be viewed as a 32 bit vector. Using the methods of Section 3.2, we get 60 inequalities formed as (7)–(10), which completely describe the propagation of division trails of $\boldsymbol{M}_{\text{Midori64}}$. Together with the compact representation of the S-box, they are really sufficient descriptions of the Midori64 round function. We have also verified this by making an exhaustive search for the solutions of the 60 inequalities and for the invertible submatrices of $\boldsymbol{M}_{\text{Midori64}}$. The number of invertible submatrices is 9,834,495 and the number of solutions of the set of inequalities is 9,834,496; the difference is caused by the all-zero solution. All the inequalities can be found in Section 11. The searching of integral distinguishers can be finished in one second by solvers such as CPLEX. As an example, we upload the VB source code for counting the number of invertible submatrices and the solutions of the 60 inequalities onto https://www.dropbox.com/s/yh7dqk11g0yvq1x/Numbers%20of%20inequalities%20and%20matrices.frm?dl=0.

For the linear layer of Midori128, we get

$$8 \times \left(\binom{4}{1} + \binom{4}{2} + \binom{4}{3} + 1 \right) = 120$$

inequalities.

4.2 7-Round integral distinguisher for Midori

Applying our method to the search for integral distinguishers, we find a 7-round integral distinguisher for Midori64 and one for Midori128. For Midori64, we let 63 bits take all values and set the division property of the input multi-set at $D^{64}_{\{\text{dfffffffffffffff}\}}$. We find that the objective function is equal to 2 after 7 rounds of encryption, which indicates that all the 64 bits satisfy a zero-sum property after 7 rounds of encryption.

For Midori128, we let 127 bits take all values and set the division property of the input multi-set at $D^{128}_{\{\text{fffffffffffffffffffffffffffffffd}\}}$. We also find that the objective function is equal to 2 after 7 rounds of encryption, which indicates that all the 64 bits satisfy zero-sum property after 7 rounds of encryption.

Note that the designers obtained only a 3.5-round integral characteristic [1]. Surprisingly, we find a 7-round integral distinguisher both for Midori64 and Midori128. This doubles the length of previously known distinguishers.

5 Application to Skinny64

In this section, we show the application of our technique to the cryptanalysis of Skinny.

5.1 Correctness of the modelling of the linear layer

Using the methods of Section 3.2, we get 60 inequalities, which completely describe the propagation of division trails of $\boldsymbol{M}_{\text{Skinny}}$. We verified that they are sufficient by checking the invertibility of all the submatrices of $\boldsymbol{M}_{\text{Skinny}}$ and by exhaustively determining all the solutions for the set of inequalities. The number of invertible submatrices is 1,185,920. The number of solutions for the set of inequalities is 1,185,921. Since the latter number includes the parasitical all-zero vector, this exercise confirms our method again. The 60 inequalities can be found in Section 12.

5.2 10-Round integral distinguisher for Skinny64

By using our method, let the division property of the input multi-set be $D^{64}_{\{\text{0fffffffffffffff}\}}$, i.e. we traverse the last 60 bits by setting the first nibble of the input to be constant and the others to be active. We find that the objective function is equal to 2 after 10 rounds of encryption, which indicates that all the 64 bits satisfy a zero-sum property after 10 rounds of encryption. The objective function is equal to 1 after 11 rounds of encryption and the experimental results show that all the 64 unit vectors occur by setting $D^{64}_{\{\text{dfffffffffffffff}\}}$. This fact indicates that there does not exist any bit satisfying a zero-sum property after 11 rounds of encryption even though we let these specific 63 bits at the input take all values.

The designers of Skinny also found an integral distinguisher which covers 10 rounds and claimed that it can be turned into a key-recovery attack on 14 rounds, without giving the details. The designers also wrote that may be the division property could be used to slightly extend those results [2]. Here, our results show there is no space for the improvement of the result on integral distinguishers by using division cryptanalysis.

We stress that the main motivation of this paper is the theoretical results on the existence of the longest integral distinguisher based on the division property. We mainly focus on proposing optimal integral distinguishers with the largest number of rounds. The key-recovery attacks on the two ciphers are skipped to make this presentation more compact.

6 Discussion

When designing a new primitive using the binary linear layer structure, the resistance against integral cryptanalysis based on division property should be considered. Theorem 2 provides a practical and subtle method for the security evaluation of block ciphers with binary linear layers against integral attacks based on the division property.

For block ciphers with a more complex matrix $\boldsymbol{M}$ such as AES, Piccolo, SM4 etc., we can also give compact descriptions of their

© The Institution of Engineering and Technology 2018

linear layers by constructing inequalities from the XOR operation of any two rows, any three rows, any four rows etc. theoretically. However, the efficiency of the method directly depends on the number of inequalities which grows with the size of the words and the blocks (as can be seen in Theorem 2). For AES, the complexity of searching their invertible submatrices is really time-consuming. About $\sim 2^{32}$ inequalities are needed to describe all of its invertible submatrices. When taking this approach, designers can use Theorem 1 to test if a division trail is valid and use Theorem 2 to construct inequalities getting rid of the fraudulent ones.

For block ciphers with a binary linear layer, the advantage of our method over the proceeding results is that we propose an appropriate number of inequalities, so that the solutions of them are exactly the division trails, without any redundancy. The number of inequalities is small enough. Although one can just use the simple branch and XOR rules given in [7, 8] to get the same rounds of distinguishers for some block ciphers, we point out that no result by division property can exceed ours. *We can prove that they are the longest trails theoretically possible based on the division property.*

We would like to mention that one can also represent the binary linear transformations of Skinny and Midori as four 4 bit linear S-boxes: each S-box acts on the variables $(x_i, x_{i+4}, x_{i+8}, x_{i+12})$ separately and independently. That is

$$\boldsymbol{M}(x_0, x_1, \ldots, x_{15}) = (S_0(x_0, x_4, x_8, x_{12}), S_1(x_1, x_5, x_9, x_{13}), S_2(x_2, x_6, x_{10}, x_{14}), S_3(x_3, x_7, x_{11}, x_{15}))\,.$$

For example, in $\boldsymbol{M}_{\text{Skinny}}$

$$S_0(x_0, x_4, x_8, x_{12}) = (x_0 + x_8 + x_{12}, x_0, x_4 + x_8, x_0 + x_8)\,.$$

For the four small S-boxes, we can generate compact MILP constraints by the method in [7]. However, by using Theorem 2, we can write the inequalities more directly and quickly. In addition, for block ciphers with a more complex matrix $\boldsymbol{M}$, Theorem 1 is a sufficient necessary condition.

7 Conclusion and future work

The integral distinguisher based on the division property is a quite recent research topic. We advanced the research in this direction by providing a compact practical description of division trails through binary diffusion layers, which are widely used in newly designed lightweight block ciphers. We apply our approach to two members of the Midori family and to Skinny64. We model the linear transformations by the smallest amount of inequalities, which is beneficial to overcome the difficulty in applying MILP in the case of binary linear layers. The method enables a practical evaluation methodology for division-property attacks in the case of binary linear layers. It has important significance in theory and practise.

Till now, there is no 5 rounds integral distinguisher found for AES, but in [17], Grassi *et al.* proposed a new structural–differential property for up to 5 rounds AES. So may be more rounds of integral distinguisher can be found if a simple and compact description for the linear layer of AES is proposed.

As a future work, we target finding a simple and compact description for block ciphers with an MDS diffusion layer such as AES, and for block ciphers such as SM4, with a linear layer composed of the XOR of circular shifts:

$$L(B) = B \oplus (B \lll 2) \oplus (B \lll 10) \oplus (B \lll 18) \oplus (B \lll 24)\,. \tag{11}$$

Since the corresponding matrix of L has some symmetry, it may lead to simplifying the description of the division trails.

8 Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grant nos. 61672330, 61602887, and the State Scholarship Fund no.201808370069 from China Scholarship Council.

9 References

[1] Banik, S., Bogdanov, A., Isobe, T., *et al.*: 'A block cipher for low energy'. Proc. ASIACRYPT, Auckland, New Zealand, November 2015, pp. 411–436
[2] Beierle, C., Jean, J., Kolbl, S., *et al.*: 'The SKINNY family of block ciphers and its low-latency variant MANTIS'. Proc. CRYPTO, Santa Barbara, USA, August 2016, pp. 123–153
[3] Ankele, R., Banik, S., Chakraborti, A., *et al.*: 'Related-key impossible-differential attack on reduced-round SKINNY'. IACR Cryptology ePrint Archive, 2016/1127, 2016
[4] Todo, Y.: 'Structural evaluation by generalized integral property'. Proc. EUROCRYPT, Sofia, Bulgaria, April 2015, pp. 287–314
[5] Todo, Y., Morii, M.: 'Bit-based division property and application to SIMON family'. Proc. Int. Conf. Fast Software Encryption, Bochum, Germany, March 2016, pp. 357–377
[6] Boura, C., Canteaut, A.: 'Another view of the division property'. Proc. CRYPTO, Santa Barbara, USA, August 2016, pp. 654–682
[7] Xiang, Z., Zhang, W., Bao, Z., *et al.*: 'Applying MILP method to searching integral distinguishers based on division property for 6 lightweight block ciphers'. Proc. ASIACRYPT, Hanoi, Vietnam, December 2016, pp. 648–678
[8] Sun, L., Wang, W., Wang, M.: 'MILP-aided bit-based division property for primitives with non-bit permutation linear layers'. IACR Cryptology ePrint Archive, 2016/811, 2016
[9] Sun, L., Wang, W., Liu, W., *et al.*: 'Automatic search of bit-based division property for ARX ciphers and word-based division property'. Proc. ASIACRYPT, Hong Kong, China, December 2017, pp. 128–157
[10] Todo, Y., Morii, M.: 'Compact representation for division property'. Proc. CANS, Milan, Italy, November 2016, pp. 19–35
[11] Sasaki, Y., Todo, Y.: 'New impossible differential search tool from design and cryptanalysis aspects – revealing structural properties of several ciphers'. Proc. EUROCRYPT, Paris, France, April 2017, pp. 185–215
[12] Mouha, N., Wang, Q., Gu, D., *et al.*: 'Differential and linear cryptanalysis using mixed-integer linear programming'. Proc. Int. Conf. INSCRYPT, Beijing, China, November 2011, pp. 57–76
[13] 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'. Proc. ASIACRYPT Kaoshiung, Taiwan, R.O.C., December 2014, pp. 158–178
[14] Knudsen, L., Wagner, D.: 'Integral cryptanalysis'. Proc. Int. Conf. Fast Software Encryption, Leuven, Belgium, February 2002, pp. 112–127
[15] Fu, K., Sun, L., Wang, M.: 'New integral attacks on SIMON', *IET Inf. Sec.*, 2017, **11**, (5), pp. 277–286
[16] Todo, Y.: 'Integral cryptanalysis on full MISTY1', *J. Cryptol.*, 2017, **30**, (3), pp. 920–959
[17] Grassi, L., Rechberger, C., Rønjom, S.: 'A new structural–differential property of 5-round AES'. Proc. EUROCRYPT, Paris, France, April 2017, pp. 289–317

10 Appendix 1. Inequalities for the S-boxes

$$\mathfrak{L}_1 = \begin{cases} x_1 + x_2 + x_3 + x_4 - y_1 - y_2 - y_3 - y_4 \geq 0 \\ -x_1 - x_2 - 2x_4 + 2y_1 + y_2 + 2y_3 + 3y_4 \geq 0 \\ -x_3 - y_1 + y_2 \geq -1 \\ x_4 + y_1 - y_2 - y_3 - y_4 \geq -1 \\ -2x_1 - x_2 - x_3 - 3x_4 - y_1 + 2y_2 + y_3 + y_4 \geq -4 \\ 3x_1 - y_1 - y_2 - y_3 - 2y_4 \geq -2 \\ -x_1 - x_2 - x_4 + y_1 - y_2 + y_3 \geq -2 \\ x_3 + 2x_4 - y_1 - y_2 - y_3 - y_4 \geq -1 \\ x_1 + x_2 - y_1 - y_2 - 2y_3 \geq -2 \\ -x_1 - x_3 + x_4 + y_1 + y_2 + 2y_3 + 2y_4 \geq 0 \\ x_1 + x_4 - y_1 - y_2 - y_3 \geq -1 \\ x_1 + 2x_2 + x_3 + x_4 - 2y_1 - 2y_3 - 2y_4 \geq -1 \\ x_i,\ y_j \text{ are binaries} \end{cases}$$

$$\mathfrak{L}_2 = \begin{cases} x_1 + x_2 + 4x_3 + x_4 - 2y_1 - 2y_2 - 2y_3 - 2y_4 \geq -1 \\ -3x_3 + y_1 + y_2 - 2y_3 + y_4 \geq -2 \\ -y_1 - y_2 + 2y_3 - y_4 \geq -1 \\ -x_1 - x_2 - x_4 + 2y_1 + 2y_2 + 2y_3 + 2y_4 \geq 0 \\ -x_2 - x_4 + y_2 + y_3 + y_4 \geq -1 \\ x_i,\ y_j \text{ are binaries} \end{cases}$$

© The Institution of Engineering and Technology 2018

$$\mathfrak{L}_3 = \begin{cases} x_1 + 4x_2 + x_3 + x_4 - 2y_1 - 2y_2 - 2y_3 - 2y_4 \geq -1 \\ 3x_4 - y_1 - y_2 - y_3 - y_4 \geq -1 \\ -x_1 - x_2 - 2x_3 - 2x_4 + 4y_1 + 4y_2 + 5y_3 + 5y_4 \geq 0 \\ 3x_3 - y_1 - y_2 - y_3 - y_4 \geq -1 \\ -4x_2 - 3x_3 - 3x_4 - y_1 - y_2 + 2y_3 + 2y_4 \geq -8 \\ -x_1 - x_4 + y_1 - y_4 \geq -2 \\ -x_1 - x_2 - x_3 - y_1 + 3y_2 - 2y_3 - y_4 \geq -4 \\ -2x_1 - x_2 - 2x_3 + y_1 - y_2 + y_4 \geq -4 \\ -2x_2 - 3x_3 - 3x_4 - 2y_1 + y_2 + y_3 + y_4 \geq -7 \\ 2y_1 - y_2 - y_3 - y_4 \geq -1 \\ x_i,\ y_j \text{ are binaries} \end{cases}$$

11 Appendix 2. Inequalities for M_{midori64}

1: $x_4 + x_8 + x_{12} - y_0 \geq 0$
2: $x_5 + x_9 + x_{13} - y_1 \geq 0$
3: $x_6 + x_{10} + x_{14} - y_2 \geq 0$
4: $x_7 + x_{11} + x_{15} - y_3 \geq 0$
5: $x_0 + x_8 + x_{12} - y_4 \geq 0$
6: $x_1 + x_9 + x_{13} - y_5 \geq 0$
7: $x_2 + x_{10} + x_{14} - y_6 \geq 0$
8: $x_3 + x_{11} + x_{15} - y_7 \geq 0$
9: $x_0 + x_4 + x_{12} - y_8 \geq 0$
10: $x_1 + x_5 + x_{13} - y_9 \geq 0$
11: $x_2 + x_6 + x_{14} - y_{10} \geq 0$
12: $x_3 + x_7 + x_{15} - y_{11} \geq 0$
13: $x_0 + x_4 + x_8 - y_{12} \geq 0$
14: $x_1 + x_5 + x_9 - y_{13} \geq 0$
15: $x_2 + x_6 + x_{10} - y_{14} \geq 0$
16: $x_3 + x_7 + x_{11} - y_{15} \geq 0$
17: $x_0 + x_4 - y_0 - y_4 \geq -1$
18: $x_1 + x_5 - y_1 - y_5 \geq -1$
19: $x_2 + x_6 - y_2 - y_6 \geq -1$
20: $x_3 + x_7 - y_3 - y_7 \geq -1$
21: $x_4 + x_8 - y_4 - y_8 \geq -1$
22: $x_5 + x_9 - y_5 - y_9 \geq -1$
23: $x_6 + x_{10} - y_6 - y_{10} \geq -1$
24: $x_7 + x_{11} - y_7 - y_{11} \geq -1$
25: $x_8 + x_{12} - y_8 - y_{12} \geq -1$
26: $x_9 + x_{13} - y_9 - y_{13} \geq -1$
27: $x_{10} + x_{14} - y_{10} - y_{14} \geq -1$
28: $x_{11} + x_{15} - y_{11} - y_{15} \geq -1$
29: $x_0 + x_8 - y_0 - y_8 \geq -1$
30: $x_1 + x_9 - y_1 - y_9 \geq -1$
31: $x_2 + x_{10} - y_2 - y_{10} \geq -1$
32: $x_3 + x_{11} - y_3 - y_{11} \geq -1$
33: $x_4 + x_{12} - y_4 - y_{12} \geq -1$
34: $x_5 + x_{13} - y_5 - y_{13} \geq -1$
35: $x_6 + x_{14} - y_6 - y_{14} \geq -1$
36: $x_7 + x_{15} - y_7 - y_{15} \geq -1$
37: $x_0 + x_{12} - y_0 - y_{12} \geq -1$
38: $x_1 + x_{13} - y_1 - y_{13} \geq -1$
39: $x_2 + x_{14} - y_2 - y_{14} \geq -1$
40: $x_3 + x_{15} - y_3 - y_{15} \geq -1$
41: $x_0 - y_4 - y_8 - y_{12} \geq -2$
42: $x_1 - y_5 - y_9 - y_{13} \geq -2$
43: $x_2 - y_6 - y_{10} - y_{14} \geq -2$
44: $x_3 - y_7 - y_{11} - y_{15} \geq -2$
45: $x_4 - y_0 - y_8 - y_{12} \geq -2$
46: $x_5 - y_1 - y_9 - y_{13} \geq -2$
47: $x_6 - y_2 - y_{10} - y_{14} \geq -2$
48: $x_7 - y_3 - y_{11} - y_{15} \geq -2$
49: $x_8 - y_0 - y_4 - y_{12} \geq -2$
50: $x_9 - y_1 - y_5 - y_{13} \geq -2$
51: $x_{10} - y_2 - y_6 - y_{14} \geq -2$
52: $x_{11} - y_3 - y_7 - y_{15} \geq -2$
53: $x_{12} - y_0 - y_4 - y_8 \geq -2$
54: $x_{13} - y_1 - y_5 - y_9 \geq -2$
55: $x_{14} - y_2 - y_6 - y_{10} \geq -2$
56: $x_{15} - y_3 - y_7 - y_{11} \geq -2$
57: $x_0 + x_4 + x_8 + x_{12} - y_0 - y_4 - y_8 - y_{12} = 0$
58: $x_1 + x_5 + x_9 + x_{13} - y_1 - y_5 - y_9 - y_{13} = 0$
59: $x_2 + x_6 + x_{10} + x_{14} - y_2 - y_6 - y_{10} - y_{14} = 0$
60: $x_3 + x_7 + x_{11} + x_{15} - y_3 - y_7 - y_{11} - y_{15} = 0$

12 Appendix 3. Inequalities for M_{Skinny}

$$M_{\text{Skinny}} = \begin{pmatrix} 1&0&0&0&0&0&0&0&1&0&0&0&1&0&0&0 \\ 0&1&0&0&0&0&0&0&0&1&0&0&0&1&0&0 \\ 0&0&1&0&0&0&0&0&0&0&1&0&0&0&1&0 \\ 0&0&0&1&0&0&0&0&0&0&0&1&0&0&0&1 \\ 1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&1&0&0&0&1&0&0&0&0&0&0&0 \\ 0&0&0&0&0&1&0&0&0&1&0&0&0&0&0&0 \\ 0&0&0&0&0&0&1&0&0&0&1&0&0&0&0&0 \\ 0&0&0&0&0&0&0&1&0&0&0&1&0&0&0&0 \\ 1&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0 \\ 0&1&0&0&0&0&0&0&0&1&0&0&0&0&0&0 \\ 0&0&1&0&0&0&0&0&0&0&1&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0&0&0&0&1&0&0&0&0 \end{pmatrix}$$

1: $x_0 + x_8 + x_{12} - y_0 \geq 0$
2: $x_1 + x_9 + x_{13} - y_1 \geq 0$
3: $x_2 + x_{10} + x_{14} - y_2 \geq 0$
4: $x_3 + x_{11} + x_{15} - y_3 \geq 0$
5: $x_0 - y_4 \geq 0$
6: $x_1 - y_5 \geq 0$
7: $x_2 - y_6 \geq 0$
8: $x_3 - y_7 \geq 0$
9: $x_4 + x_8 - y_8 \geq 0$
10: $x_5 + x_9 - y_9 \geq 0$
11: $x_6 + x_{10} - y_{10} \geq 0$
12: $x_7 + x_{11} - y_{11} \geq 0$
13: $x_0 + x_8 - y_{12} \geq 0$
14: $x_1 + x_9 - y_{13} \geq 0$
15: $x_2 + x_{10} - y_{14} \geq 0$
16: $x_3 + x_{11} - y_{15} \geq 0$
17: $x_8 + x_{12} - y_0 - y_4 \geq -1$
18: $x_9 + x_{13} - y_1 - y_5 \geq -1$
19: $x_{10} + x_{14} - y_2 - y_6 \geq -1$
20: $x_{11} + x_{15} - y_3 - y_7 \geq -1$
21: $x_0 + x_4 + x_8 - y_4 - y_8 \geq -1$
22: $x_1 + x_5 + x_9 - y_5 - y_9 \geq -1$
23: $x_2 + x_6 + x_{10} - y_6 - y_{10} \geq -1$
24: $x_3 + x_7 + x_{11} - y_7 - y_{11} \geq -1$
25: $x_0 + x_4 - y_8 - y_{12} \geq -1$
26: $x_1 + x_5 - y_9 - y_{13} \geq -1$
27: $x_2 + x_6 - y_{10} - y_{14} \geq -1$
28: $x_3 + x_7 - y_{11} - y_{15} \geq -1$
29: $x_8 - y_4 - y_{12} \geq -1$
30: $x_9 - y_5 - y_{13} \geq -1$
31: $x_{10} - y_6 - y_{14} \geq -1$

© The Institution of Engineering and Technology 2018

32: $x_{11} - y_7 - y_{15} \geq -1$
33: $x_0 + x_4 + x_{12} - y_0 - y_8 \geq -1$
34: $x_1 + x_5 + x_{13} - y_1 - y_9 \geq -1$
35: $x_2 + x_6 + x_{14} - y_2 - y_{10} \geq -1$
36: $x_3 + x_7 + x_{15} - y_3 - y_{11} \geq -1$
37: $x_{12} - y_0 - y_{12} \geq -1$
38: $x_{13} - y_1 - y_{13} \geq -1$
39: $x_{14} - y_2 - y_{14} \geq -1$
40: $x_{15} - y_3 - y_{15} \geq -1$
41: $x_4 + x_{12} - y_0 - y_4 - y_8 \geq -2$
42: $x_5 + x_{13} - y_1 - y_5 - y_9 \geq -2$
43: $x_6 + x_{14} - y_2 - y_6 - y_{10} \geq -2$
44: $x_7 + x_{15} - y_3 - y_7 - y_{11} \geq -2$
45: $x_4 - y_4 - y_8 - y_{12} \geq -2$
46: $x_5 - y_5 - y_9 - y_{13} \geq -2$
47: $x_6 - y_6 - y_{10} - y_{14} \geq -2$
48: $x_7 - y_7 - y_{11} - y_{15} \geq -2$
49: $x_4 + x_8 + x_{12} - y_0 - y_8 - y_{12} \geq -2$
50: $x_5 + x_9 + x_{13} - y_1 - y_9 - y_{13} \geq -2$
51: $x_6 + x_{10} + x_{14} - y_2 - y_{10} - y_{14} \geq -2$
52: $x_7 + x_{11} + x_{15} - y_3 - y_{11} - y_{15} \geq -2$
53: $x_0 + x_{12} - y_0 - y_4 - y_{12} \geq -2$
54: $x_1 + x_{13} - y_1 - y_5 - y_{13} \geq -2$
55: $x_2 + x_{14} - y_2 - y_6 - y_{14} \geq -2$
56: $x_3 + x_{15} - y_3 - y_7 - y_{15} \geq -2$
57: $x_0 + x_4 + x_8 + x_{12} - y_0 - y_4 - y_8 - y_{12} = 0$
58: $x_1 + x_5 + x_9 + x_{13} - y_1 - y_5 - y_9 - y_{13} = 0$
59: $x_2 + x_6 + x_{10} + x_{14} - y_2 - y_6 - y_{10} - y_{14} = 0$
60: $x_3 + x_7 + x_{11} + x_{15} - y_3 - y_7 - y_{11} - y_{15} = 0$

© The Institution of Engineering and Technology 2018