Noname manuscript No.
(will be inserted by the editor)

Block Ciphers with a Binary Diffusion Layer Against Division Cryptanalysis, How Far Can We Go?

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

Received: date / Accepted: date

Abstract In the MILP progress of searching for integral distinguishers for block ciphers, the attackers have to collect enough inequalities to describe the characters of division property propagation of round functions, including the inequalities for the S-box and the inequalities for linear layer. At present, the description of S-box is almost perfect. But the description of diffusion layers is still far from satisfactory. In this paper, in order to evaluate the security of block ciphers with binary diffusion layers against division cryptanalysis, we reveal the relation of division trails and invertible submatrices of linear transform matrix, and propose an accurate approach to model the propagation of division property of diffusion layers by the smallest amount of inequalities. The solutions of the system of inequalities are exactly the division trails of linear transformation. Hence the description is compact and optimal.

As examples of our approach, we present a 7-round integral distinguisher for both Midori64 and Midori128. That is a really inspiring advance because the designers of Midori only obtained a 3.5-round integral characteristic. For Skinny64, we find a 10-round integral distinguisher which is the same as the designers have previously found. But our result proves that 10 rounds is already the upper bound of Skinny64 when searching for integral distinguishers based on division property.

The significance of our result not only lies on that we present integral distinguishers for Midori and Skinny, but also that we shed light on how far division cryptanalysis can go on the security analysis of block ciphers with a binary diffusion layer. We can obtain the longest integral distinguishers for *block ciphers with binary diffusion layers based on division properties*, because we already get an accurate and compact description of the division properties of round function. Therefore, the technique can be used to prove security against division cryptanalysis.

Keywords Binary diffusion layer · Skinny block cipher · Midori block cipher · Division property · Integral attack

Mathematics Subject Classification (2000) 94A60

1 School of Information Science and Engineering, Shandong Normal University, Shandong, Jinan, 250014, China, E-mail: wzhang@esat.kuleuven.be ·
2 Dept. Electrical Engineering (ESAT), KU Leuven and Imec, Leuven, 3001, Belgium, E-mail: vincent.rijmen@esat.kuleuven.be

1 Introduction

Recently, in order to reduce the energy consumed by the circuit per bit in the encryption or decryption process, block cipher designers started using 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. More significantly, the main goal of Skinny is to compete with the NSA-design SIMON in terms of hardware/software performance[3]. The diffusion layer of SKINNY is an important highlight of its design strategy, with their reputation of reaching the requirements of low latency as well as fast diffusion[2]. Therefore it is of great importance to evaluate the resistance of ciphers using binary matrixes to known cryptanalysis and to give a proof of their security.

The division property [4] is a generalized 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 came up with a new approach[6] by introducing the notion of parity sets, permitting people to formulate and characterize the division property of any order in a simple way, specially for the construction of the division trails of S-boxes. At ASIACRYPT 2016, Xiang et al. proposed a method [7] to characterize 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 utilizing 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. handled the feasibility of MILP-aided division property for primitives with non-bit-permutation linear layers [8,9]. They successfully extended the MILP method to XOR based and ARX-based structures by introducing some intermediate variables among the linear layer, building $2n$ inequalities for the n- bit linear layers.

However, we found the solutions of linear inequalities in [8] contain some impossible division trails, which may eventually lead the integral-trail searching to a premature end and result in a shorter integral distinguisher. In the following, for block cipher with binary linear layer, we will give a compact description[10] of the division property propagation of linear layer.

1.1 Our Results

1. We model the propagation of the division property through a linear diffusion layer by constructing linear inequalities from the XOR operations described by the matrix of the linear layer, so that their feasible solutions exactly represent all division trails of the linear layer. Just like for S-boxes, the solutions set of the linear inequalities are equal to the sets of division trails of the linear transformation when taking the linear transformation as a super S-box. We find a way to build distinguishers for the linear layer of a block cipher and give an accurate and compact description. Thanks to the compact representation, we can accurately evaluate the propagation of the division property through binary linear layers.

2. Most importantly, we confirm that there is no potential of improvement of the result on division cryptanalysis distinguishers any more after using our method, since we already **get an accurate and tight description** of the division properties of round function. Therefore,

we propose a method to prove security against integral cryptanalysis using division property of block ciphers[5].
3. As the 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 unreasonable 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 Eurocrypt2017, Sasaki and Todo found a 7-round truncated impossible differentials for Midori128 by a automated tool[11], but there is no corresponding result for Midori64. As for Skinny64, we find a 10-round integral distinguisher for Skinny64 just the same as 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.

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

Table 1: Application results and comparing with the previous

This paper is organized as follows. In Section 2, we give some preliminaries for division property and MILP, introduce Midori and Skinny, 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 on Midori family block ciphers and show the improvements. In Section 5, we present a 10-round integral distinguisher for Skinny64. In Section 6 we give discussion about the application of our method on complex matrix and proposed the technical effects of our method prior to the proceeding ones. Finally, in Section 7, we end with a conclusion and possible future work.

2 Preliminaries and Preparations

2.1 Notations

We present our notations in Table 2.

2.2 Division Property and Division Trail

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

$$x^u = \prod_{i=0}^{n-1} x_i^{u_i}.$$

$x=(x_1,\cdots,x_n)$	An n-bit boolean vector, where x_n is the least significant bit
$wt(x)$	The hamming weight of the boolean vector x
x^T	The transposition of x
$\hat{M}$	A submatrix of M
e_i	The unit vector whose the i-th bit is 1
$wt(r_i)$	The hamming weight of the ith row of the matrix M
$\mathbb{R}$	The set of real numbers
$\mathbb{F}_2^n$	The set of all n-bit boolean vectors
$\mathbb{F}(2^n)$	A finite field of size 2^n
$\mathbb{N}$	The set of natural numbers
$\mathbb{Z}$	The set of integer numbers
t	An integer
$\mathbf{k}$	A boolean vector

Table 2: The notations used throughout the paper

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 [4,6]) A multi-set $X \subseteq \mathbb{F}_2^n$ is said to fulfill the division property 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 u that have a hamming weight less than t, i.e.

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

Notation In this paper, we consider the inputs of a transformation (for instance: a round function, an s-box, a linear transformation) travel over a subspace or a affine subspace of $\mathbb{F}_2^n$, especially the $i_1^{th},\cdots,i_t^{th}$ components of them travel from 0 to 1, and the other components are fixed. We denote the division property by $D_{\mathbf{k}}^n$, where $\mathbf{k} \in \mathbb{F}_2^n, wt(\mathbf{k}) = t$ and the $i_1^{th},\cdots,i_t^{th}$ components of $\mathbf{k}$ 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 (x_4,c_3,x_2,c_1) where x_4,x_2 vary from 0 to 1, c_3,c_1 are fixed.

Definition 2 (Division Trail [4,7,8]) 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 multiset to the block cipher has initial division property $D_{\mathbf{k}}^n$, and denote the division property after i- round propagation through f_r by $D_{\mathbb{K}_i}^n$. Thus we have the following chain of division property propagations.

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

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

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 division property through a round function of the block cipher is actually a series of transitions of the vectors. Assume that the input multiset to the S-box has division property $D_{\mathbf{k}_0}^n$, and the output multiset has division property $D_{\mathbf{k}_1}^n$, where $\mathbf{k}_0 =$

$(x_1,\cdots,x_n)$, $\mathbf{k}_1=(y_1,\cdots,y_n)$ then we call $(x_1,\cdots,x_n,y_1,\cdots,y_n)$ a division trail of this S-box [7,10]. Especially when S-box is a linear transformation, we call $(x_1,\cdots,x_n,y_1,\cdots,y_n)$ a division trail of the linear transformation. For the details of the construction of division trails, please refer to Section 5.3 in [6] or Section 2.4 in [10].

2.3 MILP and Integral Cryptanalysis

Now we introduce MILP briefly. It is a class of optimization problems derived from Linear Programming whose aim is to optimize an objective function under certain constraints. A Mixed Integer Linear Programming problem can be formally described as follows.
MILP: Find a vector $x\in\mathbb{Z}^k\times\mathbb{R}^{n-k}\subseteq\mathbb{R}^n$ with $Ax\leq b$, so that the linear function

$$c_1x_1+c_2x_2+\ldots+c_nx_n$$

is minimized (or maximized), where $(c_1,\cdots,c_n)\in\mathbb{R}^n, A\in\mathbb{R}^{m\times n}$, and $b\in\mathbb{R}^m$.

In recent years, MILP has been explicitly applied in varieties of cryptographic research areas [11,12,13,14]. We are mainly concerned about the application of the MILP method in integral cryptanalysis [15]. Integral cryptanalysis was originally proposed by Knudsen and Wagner as a dedicated attack against the block cipher Square. There are two major techniques to construct an integral characteristic; one uses the propagation characteristic of integral properties [15], and the other estimates the algebraic degree [16]. In this paper, we study the propagation of the integral property.

In the MILP progress we have to give an initial division property $D^n_{\boldsymbol{k}}$ and a stopping rule by constructing an objective function in terms of the hamming weight of the division trail.

Stopping rule. Let $D^n_{\boldsymbol{k}_i}$ be the output division property after i rounds of encryption. Let $D^n_{\boldsymbol{k}_0}$ 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 $D^n_{\boldsymbol{k}_r}$.

2.4 The Midori Block Cipher

Midori is a family of lightweight block ciphers recently published at ASIACRYPT 2015. They follow the SPN structure and have been advertised as one of the first lightweight ciphers optimized with respect to the energy consumed by the circuit per bit in the encryption or decryption operation[18]. To achieve the desired low-energy goal, several design decisions were made, like using a diffusion layer consisting of almost-MDS 4×4 binary matrices. The Midori family consists of two ciphers: Midori64 and Midori128. The block size, is 64 bits and 128 bits, 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.

1. SubCell: Apply the 4-bit S-box Sb_0 and 8-bit S-box $\mathrm{SSb}_{i(\bmod 4)}$ to each cell of Midori64 and Midori128, respectively. The truth tables of Sb_0 and Sb_1 are listed in Table 4 in Appendix A.
2. ShuffleCell: Each cell of the state is Shuffled as follows:

$$(s_0, s_1, \cdots, s_{15}) \leftarrow (s_0, s_{10}, s_5, s_{15}, s_{14}, s_4, s_{11}, s_1, s_9, s_3, s_{12}, s_6, s_7, s_{13}, s_2, s_8).$$

3. MixColumn: Multiply each column by a 4×4 matrix M_{Midori} over the finite field $\mathbb{F}(2^4)$ and $\mathbb{F}(2^8)$ correspondingly, where M_{Midori} is

$$M_{Midori} = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}.$$

Various attacks have been applied to Midori since its release. In [17] Lin et al. proposed meet in- the-middle attack. In [18] Guo et al. introduced an invariant space attack against full-round Midori64 in weak key setting. In [19,20], the authors analysis Midori128 by using 6 rounds impossible differential path. Gérault and Lafourcade proposed full-round related-key key recovery attacks on both Midori64 and Midori128 in [21]. In [11], Sasaki and Todo found a 7-round truncated impossible differential for Midori128.

2.5 The Skinny Block cipher

Skinny [2] is a family of lightweight block ciphers proposed at Crypto 2016. It adopts the SPN structure just like AES. The designers launched Skinny cryptanalysis competition after it published. Till now, a handful of third-party analysis have been published, part of the attacks such as [3,22,23]are in the related tweakey model, the others are in the impossible differential model [24].

Skinny has a variable block size of 64, 128 bits, and a key size 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:

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

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

For completeness, we list the primitive representation of M_{Skinny} as Equation (12) in Appendix B. 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 Skinny, please refer to [2].

2.6 Modeling S-boxes

In order 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 round function should satisfy the following two conditions in [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 do not influence the propagation of division property, and ShuffleCell just shuffle 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 [12]. The inequalities $\mathfrak{L}_1$ are the 5 inequalities used to describe the Midori64 Sbox $Sb_0(x)$. Their solutions are exactly the 48 division trails of $Sb_0(x)$. The inequalities $\mathfrak{L}_2$ are the 10 inequalities used to describe the Midori128 S-box $Sb_1(x)$. Their solutions are exactly the 49 division trails of $Sb_1(x)$. Since $\text{SSb}_i(x), i = 0,1,2,3$ are processed by two $\text{Sb}_1(x)$ in parallel, we need 20 inequalities for each 8-bit S-box used in Midori128. The inequalities $\mathfrak{L}_3$ in Appendix A are the 12 inequalities used to describe the Skinny S-box. Their solutions are exactly the 44 division trails of S_4.

3 A Compact Characterisation of Division Trails through Binary Linear Layers

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

In the coming we will study the sufficient and necessary conditions for a vector of length $2n$ to be a division trail of an order n linear transformation, and give a compact description of division trails through a linear transformation. First we give new insights into the division property of linear transformation. We denote the matrix of the linear transformation by $M = (a_{ij})$.

In [8], the authors deal with linear transformations in a simple modeling way by introducing some intermediate binary variables $t_{k_{(i,j)}}, 1 \leq k_{(i,j)} \leq wt(M)$ among the linear layer.

The equations used in [8] to describe the division property of linear layer are:

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

But we found that Equation (1) does not include enough inequalities. Hence in the MILP process one will get extra vectors that do not correspond to division trails. In other words, part of the solutions are not feasible, therefore their methods do not describe the division property through linear layer compactly. We describe now our method.

3.1 A 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 equally often, independent of the value of the other output bits. On the contrary, if the input of an n bit permutation can not traverse 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 k to a space of dimension 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 k. Similar to the fact,we propose the following main theorem about how the independence of input variables transfers to the output variables and how this is related to the division property of order k.

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

Proof The i-th row of M is the coefficient of the linear transformation $y_i=\sum_{k=1}^n a_{i,k}x_k$, which is the i-th 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 k-dimensional subspace onto a k-dimensional subspace. Therefore if (x,y) is one of the division trails, we have $wt(x)=wt(y)$.

"$\Longrightarrow$" By [6, Proposition 7], if (x,y) is a division trail of an order n linear transformation, then the monomial $\prod\limits_{i\in I_x} x_i$ appears in the expansion of

$$\prod_{j\in I_y} y_j = (a_{j_1,1}x_1+\cdots+a_{j_1,n}x_n)\cdots(a_{j_{wt(x)},1}x_1+\cdots+a_{j_{wt(x)},n}x_n). \tag{2}$$

In the expanded form of the Equation (2), the coefficient of $x_{i_1}\cdots x_{i_{wt(x)}}$ is a sum of terms which take exactly one coefficient as a factor from each factor of the following equation:

$$(a_{j_1,i_1}x_{i_1}+\cdots+a_{j_1,i_{wt(x)}}x_{i_{wt(x)}})\cdots(a_{j_{wt(x)},i_1}x_{i_1}+\cdots+a_{j_{wt(x)},i_{wt(x)}}x_{i_{wt(x)}}). \tag{3}$$

This results in

$$\sum_{\pi} a_{m_1,i_1} a_{m_2,i_2} \cdots a_{m_{wt(x)},i_{wt(x)}}, \tag{4}$$

where $\pi = (m_1, m_2, \cdots, m_{wt(x)}) \in \mathbb{N} \times \cdots \times \mathbb{N}$ runs over all permutation of $j_1, \cdots, j_{wt(x)}$. Equation (4) is exactly the definition of the determinant of the $wt(x) \times wt(x)$ binary matrix formed by taking the $i_1, \cdots, i_{wt(x)}$ columns and $j_1, \cdots, j_{wt(x)}$ rows from M:

$$B = \begin{pmatrix} a_{j_1,i_1} & a_{j_1,i_2} & \cdots & a_{j_1,i_{wt(x)}} \\ a_{j_2,i_1} & a_{j_2,i_2} & \cdots & a_{j_2,i_{wt(x)}} \\ \vdots & \vdots & \cdots & \vdots \\ a_{j_{wt(x)},i_1} & a_{j_{wt(x)},i_2} & \cdots & a_{j_{wt(x)},i_{wt(x)}} \end{pmatrix} \tag{5}$$

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

"$\Longleftarrow$" If $\det(B) = 1$, then the $i_1, \cdots, i_{wt(x)}$ columns of M are linearly independent, that is $x_{i_1}, \cdots, x_{i_{wt(x)}}$ are linearly independent. And similarly $y_{j_1}, \cdots, y_{j_{wt(x)}}$ are linearly independent. It means that if the input of a linear transformation has the division property x, then 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 sub-matrices of M and its division trails. The internal relation between the division trail and the matrix is the invertibility of the sub-matrix. Therefore, checking whether a vector is a division trail of a linear transformation M is equivalent to check whether the corresponding sub-matrix of M is invertible.

Example 1 To illustrate the merits of our idea over the method in [8], we take the following 4-bit linear transformation as an example: $L : (x_1, x_2, x_3, x_4) \mapsto (x_3 + x_1, x_3 + x_2 + x_1, x_3 + x_2, x_4 + x_3)$. It is an S-box with algebraic degree 1.

$$\begin{pmatrix} y_4 \\ y_3 \\ y_2 \\ y_1 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_4 \\ x_3 \\ x_2 \\ x_1 \end{pmatrix}. \tag{6}$$

According to [8], one can introduce intermediate variables $t_i, 1 \leq i \leq 9$ and get the following system of equations in binary variables:

$$\begin{cases} y_4 = t_1 + t_2 \\ y_3 = t_3 + t_6 \\ y_2 = t_4 + t_7 + t_8 \\ y_1 = t_5 + t_9 \\ x_4 = t_1 \\ x_3 = t_2 + t_3 + t_4 + t_5 \\ x_2 = t_6 + t_7 \\ x_1 = t_8 + t_9 \\ x_i, y_j, t_{k_{(i,j)}} \textit{ are binaries}. \end{cases} \tag{7}$$

Given (x_1, x_2, x_3, x_4), one can get (y_1, y_2, y_3, y_4) by the Equation (7). Table 3 lists all the 44 solutions of linear system (7), where vectors on $\mathbb{F}_2^4$ are in hexadecimal notation. However, Table 3 does not show correctly the propagation characteristic of the bit-based division property of the linear transformation L as an S-box.

k of input D_k^4	k of output D_k^4	k of input D_k^4	k of output D_k^4
0x0	{0x0}	0x8	{0x8}
0x1	{0x1,0x2}	0x9	{0x9,0xA,}
0x2	{0x2,0x4}	0xA	{0xA,0xC}
0x3	{0x3,0x5,0x6}	0xB	{0xB,0xD,0xE}
0x4	{0x1,0x2,0x4,0x8}	0xC	{0x9,0xA,0xC}
0x5	{0x5,0x6,0x9,0xA,**0x3**}	0xD	{0xD,0xE,**0xB**}
0x6	{0x3,0x5,**0x6**,0xA,0xC}	0xE	{0xD,**0xE**,0xB}
0x7	{0x7,0xB,0xD,0xE}	0xF	{0xF}

Table 3: Propagation of the bit-based division property for linear transformation L

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 stop prematurely. Now we explore how this happens. In fact, the cause of the problem is that *the left upper order 3 sub-matrix of M is noninvertible*. 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 hence it is 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, $(1110,1110) \triangleq (0xE,0xE)$ is *not* a division trail. Following the method in [8], 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.

3.2 A 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 explore a way to find 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 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 an $ns \times ns$ matrix $M = (a_{i,j}), 0 \le i,j \le ns-1$ that is derived by starting from a linear map defined by an $s \times s$ matrix $M_s \in (\mathbb{F}(2^n))^{s\times s}$ 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 M derived from a matrix M_s with coefficients $\in \{0,1\}$. Since each division trail (x,y) satisfies $wt(x) = wt(y)$, (x,y) has even hamming weight. A division trail with hamming weight $2t$ corresponds to an invertible order t submatrix of 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 nonzero entries in M. They correspond to division trails where x,y are both unit vectors. Suppose that $a_{i,k_1} =$

$\cdots = a_{i,k_{wt(r_i)}}$ are the ones in the i-th row of M. **The trails are the solutions of the set of inequalities**

$$\begin{cases} \sum_{k=0}^{ns-1} a_{i,k}x_k - y_i \geq 0, \\ y_j = 0, j \neq i. \end{cases}, \tag{8}$$

since by Equation (8), when $y_i = 1$, there is at least one of $x_k, k \in \{k_1, \cdots, k_{wt(r_i)}\}$ equals to 1. Therefore, we get $wt(r_i)$ vectors $(e_{k_1}, e_i), \cdots, (e_{k_{wt(r_i)}}, e_i) \in \mathbb{F}_2^{2ns}$. 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 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 M into n cosets. Let $\Lambda_\sigma = \{\sigma, \sigma+n, \cdots, \sigma+(s-1)n\}, 0 \leq \sigma \leq n-1$. We first consider the case $i - j \neq 0 (\bmod n)$, i.e. i and j are in different cosets. Taking into account the fact that M_s is a binary matrix over the finite field $\mathbb{F}(2^n)$, we see that when $i - j \neq 0 (\bmod n)$, the i-th row and the j-th row have no common nonzero entries. The inequalities

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

give the order two invertible submatrices containing the i-th and the j-th row.

Now we consider the case where i and j are in the same coset. We study the XOR of the i-th row and the j-th row, with $i = j (\bmod n)$. 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

$$\begin{cases} \sum_{k=0}^{ns-1} (a_{i,k} \oplus a_{j,k})x_k - y_i - y_j \geq -1, i, j \in \Lambda_\sigma \\ y_m = 0, m - \sigma \neq 0 (\bmod n). \end{cases} \tag{9}$$

to describe this phenomenon.

For example, for the third and the eleventh rows in M_{Skinny} (see Equation (12) in Appendix B), 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 component. 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 $x_3 + x_7 + x_{15} - y_3 - y_{11} \geq -1, y_i = 0, i \neq 3,\ i \neq 11$ to describe this phenomenon.

3.2.3 Describing the Trails with Hamming Weight $2t, 3 \leq t < s$

Similar to the argumentations in Subsection 3.2.2, after collecting the XOR of t special rows of M, we get the following inequalities which describe trails with hamming weight $2t$.

$$\begin{cases} \sum_{k=0}^{ns-1} (a_{i_1,k} \oplus \cdots \oplus a_{i_t,k})x_k - y_{i_1} - \cdots - y_{i_t} \geq -(t-1), \\ y_m = 0, m - \sigma \neq 0 (\bmod n), i_1, \cdots, i_t \in \Lambda_\sigma. \end{cases} \tag{10}$$

For M_{Skinny} in (12), we have $x_4 + x_{12} - y_0 - y_4 - y_8 \geq -2$. 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 0,4,8 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 $i, i+n\cdots, i+(s-1)n$ rows of M, and taking into account the compression rule of the XOR operation[7], noting that $x_i, x_{i+n}\cdots, x_{i+(s-1)n}$ are the only inputs of $y_i, y_{i+n}, \cdots, y_{i+(s-1)n}$ and $y_i, y_{i+n}, \cdots, y_{i+(s-1)n}$ only include $x_i, x_{i+n}\cdots, x_{i+(s-1)n}$, we get the following equations for some special division trails with hamming weight $2s$.

$$\begin{cases} x_i \oplus x_{i+n} \oplus \cdots \oplus x_{i+(s-1)n} = y_i \oplus y_{i+n} \oplus \cdots \oplus y_{i+(s-1)n}, i = 0, 1, \cdots, n-1, \\ y_j = 0, j - i \neq 0 (\bmod n). \end{cases} \tag{11}$$

3.3 The Necessity and Sufficiency of 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 Subsection 3.2 describe exactly the invertible submatrices of the matrix 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 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, $\{0, n, \cdots, (s-1)n\}, \{1, n+1, \cdots, (s-1)n+1\}, \cdots, \{n-1, 2n-1, \cdots, sn-1\}$. Since the rows in different cosets have no common nonzero 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 the Subsection 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 3.

Hence the four steps in Subsections 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 Modeling 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 Equations (8)–(11), which completely describe the propagation of division trails of M_{Midori64}. Together with

the compact representation of the S-box, they are really sufficient descriptions of the Midori64 round function. We have verified this again by making an exhaustive search for the solutions of the 60 inequalities and for the invertible submatrices of M_{Midori64}. The number of invertible submatrices is 9834495 and the number of solutions of the set of inequalities is 9834496; the difference is caused by the all-zero solution. All the inequalities can be found in Appendix B. The searching of integral distinguishers can be finished in one second by solvers like CPLEX. As an example, we upload 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 A 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 traverse the 63 bits by setting 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 traverse the 127 bits by setting the division property of the input multi-set at $D^{128}_{[\text{fbfffffffffffffffffffffffffffffff}]}$. 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 distinguishers 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 Modeling of the Linear Layer

Using the methods of Section 3.2, we get 60 inequalities, which completely describe the propagation of division trails of M_{Skinny}. We verified that they are sufficient by checking the invertibility of all the submatrices of M_{Skinny} and by exhaustively determining all the solutions for the set of inequalities. The number of invertible submatrices is 1185920. The number of solutions for the set of inequalities is 1185921. Since the latter number includes the parasitical all-zero vector, this exercise confirms our method again. The 60 inequalities can be found in Appendix C.

5.2 A 10-round Integral Distinguisher for Skinny64

We searched for division trails in the two different ways. By using our method, let the division property of the input multi-set be $D^{64}_{[0\text{fffffffffffffff}]}$, 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 eleven 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 eleven rounds of encryption even though we traverse the specific 63 bits at the input.

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 maybe 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 respect to division cryptanalysis, because we have given a compactness description of the round function.

We should point out that the theoretical results on the existence of the longest integral distinguisher based on division property is the motive of this paper, we mainly focus on proposing optimal integral distinguisher with the most 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 layer against integral attacks based on the division property.

For block ciphers with a more complex matrix M, such as AES, Piccolo, SM4, etc. Theoretically we can also give compact descriptions of their linear layers by constructing inequalities from the XOR operation of any two rows, any three rows, any four rows etc. But the efficiency of the method directly depends on the number of inequalities which grows with the increasing of the size of word and block(as can be seen in Theorem 2). For AES, the complexity of searching their invertible submatrices is really time consuming, about 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 binary linear layer, the advantage of our method over the results in [8] 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 simple branch and XOR rules given in [7,8] to get the same rounds distinguishers for some block ciphers, we should point out that any result by division property can not exceed ours. We can prove that they are the longest theoretically. For example, the rounds of Skinny is 10, and the round of Midori64 and Midori128 by the method in [7] are also less than or equal to ours. For any other ciphers with binary linear layer structures, our results are the optimum due to the compact representations of both S-boxes and binary linear layers. The length of any integral distinguisher based on the division property will be less than or equal to ours.

7 Conclusion and Future Work

The integral distinguisher based on the division property is quite a recent topic[25], we further the research in this direction by providing compact practical description of division trails through binary-based diffusion layers which is widely used in new designed lightweight block ciphers. We apply our approach to two members of the Midori family block ciphers and Skinny64. We can model the linear transformations by the smallest amount of inequalities that is beneficial to overcome the difficulty in applying of 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 practice.

Till now, there is no 5 rounds integral distinguisher found for AES, but in [26], the authors proposed a new structural-differential property for up to 5 rounds AES. So maybe more rounds integral distinguisher can be found if a simple and compact description for the linear layer for AES was proposed. Now, the limitation of the method in this paper is that it can only be efficiently used to give a practical evaluation methodology for division property attacks in the case of binary linear layers. As a future work, we look forward to paving a way for a simple and compact description for the block cipher with MDS diffusion layer such as AES, and block cipher SM4 which with a linear map composed of the XOR operation of circular shift of words: $L(B) = B \oplus (B \lll 2) \oplus (B \lll 10) \oplus (B \lll 18) \oplus (B \lll 24)$. Since the corresponding matrix of L has some symmetry, maybe we can simplify the description of the division trails.

References

1. Banik S., Bogdanov A., Isobe T., Shibutani K., Hiwatari H., Akishita T., Regazzoni F.: Midori: A block cipher for low energy. In: Iwata T., Cheon J. H. (eds.) ASIACRYPT 2015, Part II. LNCS, vol. 9453, pp. 411-436. Springer, Heidelberg (2015).
2. Beierle C., Jean J., Kolbl S., Leander G., Moradi A., Peyrin T., Sasaki Y., Sasdrich P., and Sim S.M.: The SKINNY family of block ciphers and its low-latency variant MANTIS. In: Robshaw M. and Katz J. (eds.) CRYPTO 2016. LNCS, vol. 9814, Part II, vol.9815, pp.123-153. Springer, Heidelberg (2016).
3. Ankele R., Banik S., Chakraborti A., List E., Mendel F., Sim S. M., Wang G.: Related-key impossible-differential attack on reduced-round SKINNY. Cryptology ePrint Archive, Report 2016/1127. https://eprint.iacr.org/ (2016).
4. Todo Y.: Structural evaluation by generalized integral property. In: Oswald E. and Fischlin M. (eds.) EUROCRYPT 2015, Part I, LNCS, vol.9056, pp.287-314. Springer, Heidelberg (2015).
5. Todo Y., Morii M.: Bit-based division property and application to Simon family. In: Peyrin T. (ed.) FSE 2016, LNCS, vol.9783, pp. 357-377. Springer, Heidelberg (2016).
6. Boura C., Canteaut A.: Another view of the division property. In: Robshaw M., Katz J. (eds.) CRYPTO 2016. LNCS, vol. 9814, Part I, pp.654-682. Springer, Heidelberg (2016).
7. Xiang Z., Zhang W., Bao Z., Lin D.: Applying MILP method to searching integral distinguishers based on division property for 6 lightweight block ciphers. In: Cheon J.H. and Takagi T. (eds.) ASIACRYPT 2016, Part I, LNCS, vol. 10031, pp.648-678, Springer, Heidelberg (2016).
8. Sun L., Wang W., Wang M.: Milp-aided bit-based division property for primitives with non-bit permutation linear layers. Cryptology ePrint Archive, Report2016/811. http://eprint.iacr.org/ (2016).
9. Sun L., Wang W., Liu W., Wang M.: Milp-aided bit-based divisionc property for ARX-based block cipher. Cryptology ePrint Archive, Report 2016/1101. http://eprint.iacr.org/ (2016).
10. Todo Y., Morii M.: Compact representation for division property. In: Foresti S., Persiano G. (eds.) CANS 2016, LNCS, vol.10052, pp.19-35, Springer, Heidelberg (2016).
11. Sasaki Y., Todo Y.: New impossible differential search tool from design and cryptanalysis aspects - revealing structural properties of several ciphers. In: Coron J.S., Nielsen J. (eds.) EUROCRYPT 2017. LNCS, vol. 10212, pp.185-215, Springer, Cham (2017).

12. Sun S., Hu L., Wang P., Qiao K., Ma X., Song L.: Automatic security evaluation and (related-key) differential characteristic search: application to Simon, Present, Lblock, DES(L) and other bit-oriented block ciphers. In: Sarkar P., Iwata T. (eds.) ASIACRYPT 2014. Part I, LNCS, vol. 8873, pp. 158-178. Springer, Heidelberg (2014).
13. Bouillaguet C., Derbez P., Fouque P. A.: Automatic search of attacks on round-reduced AES and applications. In: Rogaway P. (ed.) CRYPTO 2011. LNCS, vol.6841, pp.169-187. Springer, Heidelberg(2011).
14. Mouha N., Wang Q., Gu D., Preneel B.: Differential and linear cryptanalysis using mixed-integer linear programming. In: Wu C.-K., Young M., Lin D. (eds.) Inscrypt 2011. LNCS, vol.7537, pp.57-76. Springer, Heidelberg(2012).
15. Knudsen L. R., Wagner, D.: Integral cryptanalysis. In: Daemen J., Rijmen V. (eds.) FSE 2002. LNCS, vol. 2365, pp. 112-127. Springer, Heidelberg (2002).
16. Todo Y.: Integral cryptanalysis on full MISTY1. In: Gennaro R., Robshaw M. (eds.) CRYPTO 2015. LNCS, vol. 9215, pp.413-432. Springer, Heidelberg (2015).
17. Lin L., Wu W.: Meet-in-the-middle attacks on reduced-round Midori-64. IACR Cryptology ePrint Archive, 1165 (2015) https://eprint.iacr.org/2015/1165
18. Guo J., Jean J., Nikolic I., Qiao K., Sasaki Y., Sim S. M.: Invariant subspace attack against Midori64 and the resistance criteria for S-box designs. IACR Trans. Symmetric Cryptol. 2016(1):33-56, 2016.
19. Chen Z., Chen H., Wang X.: Cryptanalysis of Midori128 Using Impossible Differential Techniques. In: Bao F. et al. (eds.): ISPEC 2016, LNCS, vol. 10060, pp. 1C12, 2016. Springer, Heidelberg (2016).
20. Bi W., Li Z., Dong X., Wang X.: Impossible Differential Attack on Midori128 Using Rebound-like Technique. Cryptology ePrint Archive, Report 2017/286. https://eprint.iacr.org/ (2017).
21. Gérault D, Lafourcade P. Related-Key Cryptanalysis of Midori. In Dunkelman O., Sanadhya S K. (eds.) INDOCRYPT 2016. LNCS, vol. 10095, 2016: pp.287-304. Springer, Heidelberg (2016).
22. Liu G., Ghosh M., Song L.: Security analysis of SKINNY under related-tweakey settings. Cryptology ePrint Archive, Report 2016/1108. https://eprint.iacr.org/ (2016).
23. Sadeghi S., Mohammadi T., Bagheri N.: Cryptanalysis of Reduced round SKINNY Block Cipher. Cryptology ePrint Archive, Report 2016/1120. http://eprint.iacr.org/ (2016).
24. Tolba M., Abdelkhalek A., Youssef A.M.: Impossible Differential Cryptanalysis of Reduced-Round SKINNY. In Joye M., Abderrahmane Nitaj A. (eds.): AFRICACRYPT 2017: LNCS, vol. 10239, pp. 117-134.
25. Todo Y., Isobe Y., Hao Y., Meier Y. Cube Attacks on Non-Blackbox Polynomials Based on Division Property, to appear in Crypto2017.
26. Grassi L., Rechberger C. Rønjom, S.: A new structural-differential property of 5-round AES. In: Coron J.S., Nielsen J. (eds.) EUROCRYPT 2017. LNCS, vol.10211, pp. 289-317 Springer, Cham (2017).

Appendix

A Inequalities for the S-boxes

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

Table 4: Specifications of Sb_0, Sb_1 and S_4

$$\mathfrak{L}_1 = \begin{cases} x_0 + x_1 + 4x_2 + x_3 - 2y_0 - 2y_1 - 2y_2 - 2y_3 \geq -1 \\ -3x_2 + y_0 + y_1 - 2y_2 + y_3 \geq -2 \\ -y_0 - y_1 + 2y_2 - y_3 \geq -1 \\ -x_0 - x_1 - x_3 + 2y_0 + 2y_1 + 2y_2 + 2y_3 \geq 0 \\ -x_1 - x_3 + y_1 + y_2 + y_3 \geq -1 \\ x_i, y_j \text{ are binaries} \end{cases}$$

$$\mathfrak{L}_2 = \begin{cases} x_0+4x_1+x_2+x_3-2y_0-2y_1-2y_2-2y_3 \geq -1 \\ 3x_3-y_0-y_1-y_2-y_3 \geq -1 \\ -x_0-x_1-2x_2-2x_3+4y_0+4y_1+5y_2+5y_3 \geq 0 \\ 3x_2-y_0-y_1-y_2-y_3 \geq -1 \\ -4x_1-3x_2-3x_3-y_0-y_1+2y_2+2y_3 \geq -8 \\ -x_0-x_3+y_0-y_3 \geq -2 \\ -x_0-x_1-x_2-y_0+3y_1-2y_2-y_3 \geq -4 \\ -2x_0-x_1-2x_2+y_0-y_1+y_3 \geq -4 \\ -2x_1-3x_2-3x_3-2y_0+y_1+y_2+y_3 \geq -7 \\ 2y_0-y_1-y_2-y_3 \geq -1 \\ x_i, y_j \text{ are binaries} \end{cases}$$

$$\mathfrak{L}_3 = \begin{cases} x_0+x_1+x_2+x_3-y_0-y_1-y_2-y_3 \geq 0 \\ -x_0-x_1-2x_3+2y_0+y_1+2y_2+3y_3 \geq 0 \\ -x_2-y_0+y_1 \geq -1 \\ x_3+y_0-y_1-y_2-y_3 \geq -1 \\ -2x_0-x_1-x_2-3x_3-y_0+2y_1+y_2+y_3 \geq -4 \\ 3x_0-y_0-y_1-y_2-2y_3 \geq -2 \\ -x_0-x_1-x_3+y_0-y_1+y_2 \geq -2 \\ x_2+2x_3-y_0-y_1-y_2-y_3 \geq -1 \\ x_0+x_1-y_0-y_1-2y_2 \geq -2 \\ -x_0-x_2+x_3+y_0+y_1+2y_2+2y_3 \geq 0 \\ x_0+x_3-y_0-y_1-y_2 \geq -1 \\ x_0+2x_1+x_2+x_3-2y_0-2y_2-2y_3 \geq -1 \\ x_i, y_j \text{ are binaries} \end{cases}$$

B 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: $x_3 + x_7 + x_{11} + x_{15} - y_3 - y_7 - y_{11} - y_{15} = 0$

C Inequalities for M_{Skinny}

$$M_{\text{Skinny}} = \begin{pmatrix}
1&0&0&0&0&0&0&0&1&0&0&0&1&0&0&0\\
0&1&0&0&0&0&0&0&0&1&0&0&0&1&0&0\\
0&0&1&0&0&0&0&0&0&0&1&0&0&0&1&0\\
0&0&0&1&0&0&0&0&0&0&0&1&0&0&0&1\\
1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&1&0&0&0&1&0&0&0&0&0&0&0\\
0&0&0&0&0&1&0&0&0&1&0&0&0&0&0&0\\
0&0&0&0&0&0&1&0&0&0&1&0&0&0&0&0\\
0&0&0&0&0&0&0&1&0&0&0&1&0&0&0&0\\
1&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0\\
0&1&0&0&0&0&0&0&0&1&0&0&0&0&0&0\\
0&0&1&0&0&0&0&0&0&0&1&0&0&0&0&0\\
0&0&0&1&0&0&0&0&0&0&0&1&0&0&0&0
\end{pmatrix} \tag{12}$$

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: $x_{10} - y_6 - y_{14} \geq -1$
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: $x_3 + x_7 + x_{11} + x_{15} - y_3 - y_7 - y_{11} - y_{15} = 0$