Division Cryptanalysis of Block Ciphers with a Binary Diffusion Layer

Wenying Zhang[1,2] and Vincent Rijmen[2]

[1] School of Information Science and Engineering,
Shandong Normal University, Jinan, China,
[2] Dept. Electrical Engineering (ESAT), KU Leuven and Imec, Leuven, Belgium
{wzhang,vincent.rijmen}@esat.kuleuven.be

Abstract. In this paper, we propose an accurate security evaluation methodology for block ciphers with a binary diffusion layers against division cryptanalysis. We illustrate the division property by the independence of variables, and exploit a one-to-one mapping between division trails and invertible sub-matrices. We give a new way to model the propagation of division property of linear diffusion layers by the smallest amount of inequalities which are generated from linear combinations of row vectors of the diffusion matrix. The solutions of these inequalities are exactly the division trails of linear transformation. Hence the description is compact and optimal.
As applications of our methodology, we first present a 10-round integral distinguisher for Skinny, proposed at CRYPTO 2016 which is of one round more than that found by using the previous method. For Midori, proposed at ASIACRYPT 2015, the designers have obtained a 3.5-round integral characteristic. Surprisingly, we find 7-round integral distinguishers both for Midori64 and Midori128.
Most importantly, we obtain the longest integral distinguishers for block ciphers with a binary diffusion layer. It seems that any more improvement of this kind of integral distinguishers using the division property is impossible. Therefore, the technique can be used to prove security against division cryptanalysis, and we can hopefully expect it to become a useful technique for designers.

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

1 Introduction

Recently, in order to optimize 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. With their reputation of reaching the requirements of low latency in an unrolled implementation as well as fast diffusion[2], 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 [12] is a generalized integral property initially proposed by Todo at EUROCRYPT 2015. At FSE 2016, Todo and Morri proposed the bit-based division property and applied it to find a 14-round integral distinguisher for SIMON32[13]. At CRYPTO 2016, Christina Boura and Anne Canteaut came up with a new approach[4] 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

[15] 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 [9, 8]. 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, based on which they claimed their inequalities to be "sufficient". However, we found that the solutions of linear inequalities in [9] contain some impossible division trails, which may eventually lead the integral-trail searching to come to a premature end and result in a shorter integral distinguisher.

In the MILP progress of searching for integral distinguishers based on division property, we first give an initial division property which stands for the position of active bits and the fixed bits. Secondly we collect enough inequalities to describe the characters of division property propagates of round functions, including the inequalities for the S-box and the inequalities for liner layer. At present, the description of S-box is almost perfect[15, 9, 8]. But the description of general diffusion layers is still far from satisfactory. In this paper, we partially cover this gap by studying the division property of binary linear diffusion layers.

1.1 Our Contributions

This work aims at giving a compact inequality-based [14] description of the division property of a binary linear diffusion layer. 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 solution sets of the set of 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.

1. We link the division property to the independence of variables and give a novel model for the propagation of a division property propagation through a linear layer, which is a new method to study the integral characteristics of the linear layer of a block cipher. We find that there is a one-to-one map from the vectors in the division trails of linear transformation to invertible sub-matrices of the linear transformation matrix M, and we can give a simple description of the invertibility of the sub-matrices by some inequalities. It essentially relates to the linear combination of the rows and the columns of the sub-matrix in M. In final analysis, the division property represents the independence of variables, which was also the original intention of its proposition.

2. As for the applications of our methodology, we propose compact representations for the division trails through the linear layer of the newly proposed block ciphers Skinny and Midori. We describe exactly all division trails of their linear layers without any extra unreasonable one. We find a 10-round integral distinguisher for Skinny, which is one more round than the integral distinguisher found by using the method in [9]. The designers of Skinny mention in their security analysis that the division property can probably be used to slightly extend their results. Our results, however, show that there is no feasiblity of any more improvement of the results on the integral distinguisher attack.

3. As for Midori64 and Midori128, the designers have obtained a 3.5-round integral characteristic. Surprisingly, we find 7-round characteristics, twice what the former method achieves, both for Midori64 and Midori128.
4. Most importantly, we can claim that there is no potential of improvement of the result on the attack by distinguishers after using our method, since we already get an accurate description of all the division trails of the round functions. Therefore, we propose a method to prove security against integral cryptanalysis using division property for block ciphers with a binary linear diffusion layer [13].

The subsequent part of this paper is organized as follows. In Section 2, we give some preliminaries for division property and MILP, introduce Skinny and Midori 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 trails of binary linear layers. In Section 4, as an application of methods in Section 3, we present a 10-round integral distinguisher for Skinny64. In Section 5, we apply the method on Midori family block ciphers and show the improvements. Finally, conclusions are drawn in Section 6.

2 Preliminaries and Preparations

2.1 Notations

We present our notations in Table 1.

Table 1: The notations used throughout the paper

$x = (x_1, \cdots, x_n)$	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 i^{th} 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

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

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

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 [12, 4])**.** A set $X \subseteq F_2^n$ has the division property D_k^n for some $1 \leq k \leq n$, 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 k, i.e.

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

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

Definition 2 (Division Trail [12, 9])**.** Let f_r be the round function of an iterated block cipher. Assume that the input multiset to the block cipher has initial division property D_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.

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

Moreover, for any vector k_i^* in $\mathbb{K}_i (i \geq 1)$, there must exist an vector k_{i-1}^* in $\mathbb{K}_{i-1}$ such that k_{i-1}^* can propagate to k_i^* by division property propagation rules. Furthermore, for $(k_0, k_1, \cdots, k_r) \in \mathbb{K}_0 \times \mathbb{K}_1 \times \cdots \times \mathbb{K}_r$, if k_{i-1} can propagate to k_i for all $i \in \{1, 2, \cdots, r\}$, we call $(k_0, k_1, \cdots, k_r)$ an r-round division trail.

Assume that the input multiset to the S-box has division property $D_{\mathbf{k}_1}^n$, and the output multiset has division property $D_{\mathbf{k}_2}^n$, where $\mathbf{k}_1 = (x_1, \cdots, x_n)$, $\mathbf{k}_2 = (y_1, \cdots, y_n)$ then we call $(x_1, \cdots, x_n, y_1, \cdots, y_n)$ a division trail of this S-box [15].

The propagation of division property through a round function of the block cipher is actually a series of transitions of the vectors. In a typical SPN cipher, the first transition is from the input of the round function to the outputs of the S-boxes, which are the inputs of the linear layer. The second transition is through the linear layer and its output equals the output of one round of the block cipher.

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

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

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 cryptographic research [10, 3, 7]. We are mainly concerned about the application of the MILP method in integral cryptanalysis. Roughly speaking, the integral attack is a cryptanalytic technique used to discover a zero-sum property of the ciphertext. Attackers first prepare N chosen plaintexts and encrypt them for R rounds. If the XOR of all encrypted texts becomes 0, we can say that the cipher has an R-round integral characteristic with N chosen plaintexts. Finally, they analyze the entire cipher by using the integral characteristic. There are two major techniques to construct an integral characteristic; one uses the propagation characteristic of integral properties [6], and the other estimates the algebraic degree [11]. In this paper, we study the propagation of the integral property.

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

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

A linear inequality system $Ax \leq b$ will be adopted to describe division property propagations. So we have to construct a linear inequality system whose solutions exactly represent all division trails. The following two conditions come from [15] which are sufficient for block ciphers using a diffusion layer consisting of bit permutations:

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 does not contain any impossible division trail.

2.4 The Skinny Block cipher

Skinny [2] is a family of lightweight block ciphers proposed by Christof Beierle et al. at Crypto 2016. It adopts the SPN structure just like AES. Skinny has a variable block size of 64, 128 bits, and a key size of 64, 192 or 256 bits. In this paper, we 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: Applying the 4-bit S-box S_4 (see the Append) on each nibble.
2. AddConstants and AddRoundKey(AK): XOR the state with constant and subkey.
3. ShiftRow: The i-th row is shifted by i bytes to the right, $i = 0, 1, 2, 3$.
4. MixColumn: Multiplying each column by a constant 4×4 matrix M_{Skinny} over the field F_{2^4} , where

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

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. We omit the key schedule of Skinny. For more details about Skinny, please refer to [2].

2.5 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. 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 [5].

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. Similar to Skinny and AES, the internal state of Midori is represented as an array of 4×4 cells $s_i, 0 \leq i \leq 15$, where 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: Applying the 4-bit S-box Sb_0 or 8-bit S-box $\mathrm{SSb}_{i(\bmod 4)}$ on each cell of Midori64 respectively Midori128. Here the $\mathrm{SS}b_i$ are 8-bit S-boxes consisting of two 4-bit S-boxes processed in parallel. The truth tables of Sb_0 and Sb_1 are listed in Table 3 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: Multiplying each column by a 4×4 matrix M_{Midori} over the finite field F_{2^4} and F_{2^8} correspondingly, where M_{Midori} is

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

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 for the round function. Since XORing with constants, subkeys and shift rows do not influence the division property, 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 [4] to generate the propagation table of the S-box. After that, just as what has been introduced in [15], 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 [10]. The inequalities $\mathfrak{L}_1$ 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. The inequalities $\mathfrak{L}_2$ are the 5 inequalities used to describe the Midori64 Sbox $Sb_0(x)$. Their solutions are exactly the 48 division trails of $Sb_0(x)$, where $(x_1, x_2, x_3, x_4) \rightarrow (y_1, y_2, y_3, y_4)$ denotes a division trail. The inequalities $\mathfrak{L}_3$ are the 10 inequalities used to describe the Midori128 Sbox $Sb_1(x)$. Their solutions are exactly the 49 division trails of $Sb_1(x)$. Since $\mathrm{SSb}_i(x), i = 0, 1, 2, 3$ are processed by two $\mathrm{Sb}_1(x)$ in parallel, we need 20 inequalities for each 8-bit Sbox used in Midori128.

3 A Compact Characterisation of Division Trails through Binary Linear Layers

Modeling the linear layer is a distinctive step of MILP. For diffusion layers consisting of bit permutations, the methods described in [9] satisfy the two conditions mentioned at the end of Section 2.3. However, for more general linear diffusion layers, their methods do 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, hence their method does not describe the division trails through linear layers perfectly.

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 [9], 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 [9] to describe the linear layer are:

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

We describe now our method.

3.1 A Compact Theoretical Description

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 is run 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. As a direct consequence of this 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 transform. 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 transform if and only if the order $wt(x)$ sub-matrix whose rows indices are taken from I_x and columns indices are taken from I_y is invertible.*

Proof. First we review the calculation method of division trails of an S-box. The i-th row of M is the coefficient of the linear transform $y_j = \Sigma_{k=1}^{n} a_{j,k} x_k$, which is the i-th component of the linear transformation, corresponding to the list of all monomials with degree one in the ANF of $x \mapsto y_j(x)$. Since the linear transform is invertible, it maps a k-dimensional subspace onto a k-dimensional subspace. Therefore $wt(x) = wt(y)$.

"$\Longrightarrow$" By [4, Proposition 7], if (x, y) is one of the division trails of an order n linear transformation, then the monomial $\prod\limits_{i \in I_x} x_i$ appears in the expansion of

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

By combinatorics, the coefficient of $x_{i_1} \cdots x_{i_{wt(x)}}$ is a sum of terms where each term takes exactly one coefficient 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

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

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

$j_1, \cdots, j_{wt(x)}$ columns from M.

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

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

"$\Longleftarrow$" If $\det(B) = 1$, then the $j_1, \cdots, j_{wt(x)}$ rows of M are linearly independent, which means that $x_{j_1}, \cdots, x_{j_{wt(x)}}$ are linearly independent. And similarly $y_{i_1}, \cdots, y_{i_{wt(x)}}$ are linearly independent. Hence (x, y) is in the division table of M. □

The internal relation between the division trail and the matrix is the invertibility of the sub-matrix. In other words, if one sub-matrix whose rows indices are taken from I_x and columns indices are taken from I_y is invertible, then the vector of length $2n$ whose $i_1, i_2, \cdots, i_r, n + j_1, n + j_2, \cdots, n + j_r$'th coordinates are 1, is a division trail.

According to Theorem 1, 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. In order to get all the division trails, it is sufficient to find all the invertible sub-matrices of M.

Example 1. To illustrate the merits of our idea over the method in [9], 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 [9], one can introduce intermediate variables $t_i, 1 \leq i \leq 9$ and get the following system of equations in binary variables:

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

Table 2 lists all the 44 solutions of (7), where vectors on $\mathbb{F}_2^4$ are in hexadecimal notation. However, Table 2 does not show correctly the propagation characteristic of the bit-based division property of the linear transformation L as an S-box.

Indeed, the three vectors shown in bold are superfluous; the correct propagation of division property has all vectors excluding the three vectors in bold. It must be noted that, in some time, these three 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 appears in this expression, it appears twice, and hence it is cancelled out. It follows that $y_4y_3y_2$ should not be include in the division table corresponding to the input $x_4x_3x_2$. That is, $(1110, 1110) \triangleq (0\text{xE}, 0\text{xE})$ is *not* a division

Table 2: Propagation of the bit-based division property for 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
0x0	{0x0}	0x08	{0x0}
0x1	{0x1,0x2}	0x09	{0x90xA,}
0x2	{0x2,0x04}	0x0A	{0xA,0xC}
0x3	{0x3,0x05,0x06}	0x0B	{0xB,0xD,0xE}
0x4	{0x1,0x02,0x04,0x08}	0x0C	{0x9,0xA,0xC}
0x5	{0x5,0x06,0x09,0x0A,**0x03**}	0x0D	{0xD,0xE,**0xB**}
0x6	{0x3,0x5,0x6,0xA,0xC}	0x0E	{0xD,**0xE**,0xB}
0x7	{0x7,0xB,0xD,0xE}	0x0F	{0xE}

trail. Following the method in [9], the division trail is computed by verifying whether a monomial appears *at least once* in the calculation progress of the ANF, without taking into account possible cancellation. There is a one to one corresponding between the invertible sub-matrices of M and its division trails.

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, we only need to find all the invertible submatrices of M. Each of them corresponds to a division trail. Now the key point is to have a way to describe the character of the invertible sub-matrices. Inspired by the way of representing the division trails of S-box as linear inequalities in [15], we want to find a practical way to describe the division property of linear transformation by inequalities. It tuirns out that the inequalities which are deduced from the linear transformation $y^T = Mx^T$ can describe the hamming weight 2 division trails. Similarly, we obtain a practical compact description for all division trails.

In the following we consider an $ns \times ns$ matrix $M = (a_{i,j}), 0 \leq i, j \leq ns-1$ 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 matrices M derived from a matrix M_s with coefficients $\in \{0, 1\}$.

Since each division trail (x, y) satisfies $wt(x) = wt(y)$, all vectors (x, y) have 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 by Inequalities

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

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

since by Equation (8), when $y_i = 1$, there is at least one of $x_k, k \in \{k_1, \cdots, k_{wt(r_i)}\}$ that equals 1. Therefore, we get $wt(r_i)$ vectors $(e_{k_1}, e_i), \cdots, (e_{k_{wt(r_i)}}, e_i) \in \mathbb{F}_2^{ns}$. They are all the division trails with hamming weight 2.

3.2.2 Describing the Trails with Hamming Weight 4 by Inequalities

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 $i-j \neq 0(\mathrm{mod}n)$, i.e. i and j are in a different coset. Taking into account the fact that M_s is a binary matrix over the finite field F_{2^n}, we see that when $i-j \neq 0(\mathrm{mod}n)$, the i-th row and the j-th row have no common nonzero entries. The inequalities

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

describe all 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 $i = j(\mathrm{mod}n)$. 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

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

to describe this phenomenon.

For example, for the third and the eleventh rows in M_{Skinny} (see (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 $2t, 3 \leq t < s$ by Inequalities

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 $2t$.

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

For example, 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 Inequalities

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[15], 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 $2s$.

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

3.3 The Necessity and Sufficiency of Preceeding 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 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 (8)–(11). 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 three bold vectors in Table 2.

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 Skinny64

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

4.1 Correctness of the Modeling of the Linear Layer

Each division trail of the linear transforation of Skinny can be viewed as a 32-bit vector. Uisng 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_{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. The 60 inequalities can be found in Appendix B.

4.2 A 10-round Integral Distinguisher for Skinny64

In order to compare the experimental results using our method with that of [9], we searched for division trails in the two different ways. By using our method, let the division property of the input multi-set be $D^{64}_{[0\text{fffffffffff}]}$, i.e., we traverse the last 60 bits by setting the first nibble of the input to be constant and the others 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 $D^{64}_{[\text{dfffffffffff}]}$. 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.

However, by the method in [9], we can only get a 9-round integral distinguisher. Although we traverse 63 bits of input, i.e., $D^{64}_{[\text{dfffffffffff}]}$, the objective function is equal to 1 after ten rounds of encryption. That means that our method finds more trails than that in [9].

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 [2]. The designers also wrote that maybe the division property could be used to slightly extend those results. Here our results show there is no space for improvement of the result on this type of distinguishers.

5 Application to Midori

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

5.1 Modeling the Linear Layer

For the linear layer in Midori, by taking account in the nonzero entries in each row, and the XOR operation of two rows, three rows and four rows respectively, we get 60 inequalities, which completely describe the propagation of division trails. 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 again caused by the all-zero solution. Hence we confirmed again our method. All the inequalities can be found in Appendix B. For the linear layer of Midori128, we get $\binom{32}{1} + 8 \times \binom{4}{2} + 8 \times \binom{4}{3} + 4 = 120$ inequalities.

5.2 A 7-round Integral Distinguisher for Midori

There are some guidelines for the arrangement of the fixed bits at the input (i.e. the initial division property) of MILP to maximise the length of the division trail. Firstly, the fixed bits at the input should be selected such that the hamming weight of the output of the S-box is as large as possible. For example for Sb_0, when the input is 1101, the division trail is $[11010011], [11010110], [11011010]$, but when the input is 1011, the division trail is $[10110001], [10110100]$, $[10111000]$, so we prefer 1101 as the initial division property of one S-box in order to avoid a fast reduction of the hamming weight.

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 $D^{64}_{[\text{dfffffffffff}]}$. 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 $D^{128}_{[\mathtt{fbffffffffffffffffffffffffffff}]}$. 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 is double the length of previously known distinguishers.

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 an efficient 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 the matrix in AES, Piccolo, etc., We can construct inequalities by the XOR operation of two rows, three rows, four rows etc. The combination should contain as many rows as possible to get rid of fraudulent trails with low hamming weights. We can also get some inequalities from the inverse transformation of the linear layer.

When taking this approach, designers can use Theorem 1 to test if a division trail is valid and use Theorem 2 to construct inequalities get rid of the fraudulent ones. How many inequalities are required depens on the intermediate states of the MILP processing procedure.

7 Conclusions

In this paper, we propose a new and rigorous description of integral characteristics for the linear layer of block ciphers. We construct inequalities whose solutions correspond one-on-one to the division trails of binary linear transformations. Combined with the accurate description of the S-box, we can get a compact representation of the round function. Hence we give a new security evaluation methodology for block ciphers with binary linear layers against the integral attack.

As an application, we get a ten-round integral characteristics, which is one more round than by using the method in [9]. We find 7-round integral distinguishers for Midori64 and Midori128, twice the length of previously known distinguishers.

We claim that for SPN block ciphers with binary linear layers there exist no longer integral trails based on the division property than those that are found our methods. Therefore we give a security proof against the integral attack for SPN block cipher using binary linear diffusion layers. Perhaps the compact method can also facilitate the search for impossible difference paths.

References

[1] Subhadeep Banik, Andrey Bogdanov, Takanori Isobe, Kyoji Shibutani, Harunaga Hiwatari, Toru Akishita, and Francesco Regazzoni. Midori: A block cipher for low energy. In Tetsu Iwata and Jung Hee Cheon, editors, *Advances in Cryptology - ASIACRYPT 2015 - 21st International Conference on the Theory and Application of Cryptology and Information Security, Auckland, New Zealand, November 29 - December 3, 2015, Proceedings, Part II*, volume 9453 of *Lecture Notes in Computer Science*, pages 411–436. Springer, 2015.

[2] Christof Beierle, Jérémy Jean, Stefan Kölbl, Gregor Leander, Amir Moradi, Thomas Peyrin, Yu Sasaki, Pascal Sasdrich, and Siang Meng Sim. The SKINNY family of block ciphers and its low-latency variant MANTIS. In Matthew Robshaw and Jonathan Katz, editors, *Advances in Cryptology - CRYPTO 2016 - 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part II*, volume 9815, pages 123–153. Springer, 2016.

[3] Charles Bouillaguet, Patrick Derbez, and Pierre-Alain Fouque. Automatic search of attacks on round-reduced AES and applications. In Phillip Rogaway, editor, *Advances in Cryptology - CRYPTO 2011 - 31st Annual Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2011. Proceedings*, volume 6841 of *Lecture Notes in Computer Science*, pages 169–187. Springer, 2011.

[4] Christina Boura and Anne Canteaut. Another view of the division property. In Matthew Robshaw and Jonathan Katz, editors, *Advances in Cryptology - CRYPTO 2016 - 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part I*, volume 9814 of *Lecture Notes in Computer Science*, pages 654–682. Springer, 2016.

[5] Jian Guo, Jérémy Jean, Ivica Nikolic, Kexin Qiao, Yu Sasaki, and Siang Meng Sim. Invariant subspace attack against midori64 and the resistance criteria for s-box designs. *IACR Trans. Symmetric Cryptol.*, 2016(1):33–56, 2016.

[6] Lars R. Knudsen and David Wagner. Integral cryptanalysis. In Joan Daemen and Vincent Rijmen, editors, *Fast Software Encryption, 9th International Workshop, FSE 2002, Leuven, Belgium, February 4-6, 2002, Revised Papers*, volume 2365 of *Lecture Notes in Computer Science*, pages 112–127. Springer, 2002.

[7] Yu Sasaki and Yosuke Todo. New impossible differential search tool from design and cryptanalysis aspects. Cryptology ePrint Archive, Report 2016/1181, 2016. http://eprint.iacr.org/2016/1181.

[8] Ling Sun, Wei Wang, Wei Liu, and Meiqin Wang. Milp-aided bit-based division property for arx-based block cipher. Cryptology ePrint Archive, Report 2016/1101, 2016. http://eprint.iacr.org/2016/1101.

[9] Ling Sun, Wei Wang, and Meiqin Wang. Milp-aided bit-based division property for primitives with non-bit-permutation linear layers. Cryptology ePrint Archive, Report 2016/811, 2016. http://eprint.iacr.org/2016/811.

[10] Siwei Sun, Lei Hu, Peng Wang, Kexin Qiao, Xiaoshuang Ma, and Ling Song. Automatic security evaluation and (related-key) differential characteristic search: Application to simon, present, lblock, DES(L) and other bit-oriented block ciphers. In Palash Sarkar and Tetsu Iwata, editors, *Advances in Cryptology - ASIACRYPT 2014 - 20th International Conference on the Theory and Application of Cryptology and Information Security, Kaoshiung, Taiwan, R.O.C., December 7-11, 2014. Proceedings, Part I*, volume 8873 of *Lecture Notes in Computer Science*, pages 158–178. Springer, 2014.

[11] Yosuke Todo. Integral cryptanalysis on full MISTY1. In Rosario Gennaro and Matthew Robshaw, editors, *Advances in Cryptology - CRYPTO 2015 - 35th Annual Cryptology Conference, Santa Barbara, CA, USA, August 16-20, 2015, Proceedings, Part I*, volume 9215 of *Lecture Notes in Computer Science*, pages 413–432. Springer, 2015.

[12] Yosuke Todo. Structural evaluation by generalized integral property. In Elisabeth Oswald and Marc Fischlin, editors, *Advances in Cryptology - EUROCRYPT 2015 - 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Sofia, Bulgaria, April 26-30, 2015, Proceedings, Part I*, volume 9056 of *Lecture Notes in Computer Science*, pages 287–314. Springer, 2015.

[13] Yosuke Todo and Masakatu Morii. Bit-based division property and application to simon family. In Thomas Peyrin, editor, *Fast Software Encryption - 23rd International Conference, FSE 2016, Bochum, Germany, March 20-23, 2016, Revised Selected Papers*, volume 9783 of *Lecture Notes in Computer Science*, pages 357–377. Springer, 2016.

[14] Yosuke Todo and Masakatu Morii. Compact representation for division property. In Sara Foresti and Giuseppe Persiano, editors, *Cryptology and Network Security - 15th International Conference, CANS 2016, Milan, Italy, November 14-16, 2016, Proceedings*, volume 10052 of *Lecture Notes in Computer Science*, pages 19–35, 2016.

[15] Zejun Xiang, Wentao Zhang, Zhenzhen Bao, and Dongdai Lin. Applying MILP method to searching integral distinguishers based on division property for 6 lightweight block ciphers. In Jung Hee Cheon and Tsuyoshi Takagi, editors, *Advances in Cryptology - ASIACRYPT 2016 - 22nd International Conference on the Theory and Application of Cryptology and Information Security, Hanoi, Vietnam, December 4-8, 2016, Proceedings, Part I*, volume 10031 of *Lecture Notes in Computer Science*, pages 648–678, 2016.

A Inequalities for the Midori S-boxes

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

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

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

$$\mathfrak{L}_3 = \begin{cases} x_0 + 4x_1 + x_2 + x_3 - 2y_0 - 2y_1 - 2y_2 - 2y_3 \geq -1 \\ 3x_3 - y_0 - y_1 - y_2 - y_3 \geq -1 \\ -x_0 - x_1 - 2x_2 - 2x_3 + 4y_0 + 4y_1 + 5y_2 + 5y_3 \geq 0 \\ 3x_2 - y_0 - y_1 - y_2 - y_3 \geq -1 \\ -4x_1 - 3x_2 - 3x_3 - y_0 - y_1 + 2y_2 + 2y_3 \geq -8 \\ -x_0 - x_3 + y_0 - y_3 \geq -2 \\ -x_0 - x_1 - x_2 - y_0 + 3y_1 - 2y_2 - y_3 \geq -4 \\ -2x_0 - x_1 - 2x_2 + y_0 - y_1 + y_3 \geq -4 \\ -2x_1 - 3x_2 - 3x_3 - 2y_0 + y_1 + y_2 + y_3 \geq -7 \\ 2y_0 - y_1 - y_2 - y_3 \geq -1 \\ x_i, y_j \text{ are binaries} \end{cases}$$

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

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