Wide Trail Design Strategy for Binary MixColumns

Enhancing Lower Bound of Number of Active S-boxes

Yosuke Todo(✉) and Kazumaro Aoki

NTT Secure Platform Laboratories, Tokyo, Japan
todo.yosuke@lab.ntt.co.jp

Abstract. AES is one of the most common block ciphers and many AES-like primitives have been proposed. Recently, many lightweight symmetric-key cryptographic primitives have also been proposed. Some such primitives require the diffusion using element-wise XORs, which are called *binary matrices* in this paper, rather than that using MDS matrices because the element-wise XOR is efficiently implemented in a lightweight environment. However, since the branch number of binary matrices is generally lower than that of MDS matrices, such primitives require more rounds to guarantee security against several cryptanalyses. In this paper, we focus on binary matrices and discuss useful cryptographic properties of binary matrices. Specifically, we focus on AES-like primitives with *binary MixColumns*, whose output is computed using a binary matrix. One of the benefit of AES-like primitives is that four rounds guarantee $\mathcal{B}^2$ differentially and linearly active S-boxes, where $\mathcal{B}$ denotes the branch number of the matrix. We argue that there is a binary MixColumns in which the lower bound of the number of active S-boxes is more than $\mathcal{B}^2$ in the 4-round characteristic. For some binary matrices, the lower bound is improved from $\mathcal{B}^2$ to $\mathcal{B}(\mathcal{B}+2)$.

Keywords: Differential attack · Linear attack · Active S-box · AES-like primitive · MDS · Binary MixColumns

1 Introduction

Many symmetric key cryptographic primitives, e.g., block ciphers, compression functions of hash functions, and core functions of authenticated encryptions, have been proposed. Specifically, AES [1] is one of the most common block ciphers. The state is represented as a 4×4 matrix whose elements take 8-bit values. After AES was standardized by NIST, many AES-like primitives have been proposed [2,5,10,17,19–21]. Their state is represented as an $n \times m$ matrix, and its elements take not only 8-bit values. We call such primitives (n, m)-AES-like primitives. PHOTON [19] can be considered as $(5, 5)$, $(6, 6)$, ..., $(8, 8)$-AES-like primitives, and PRIMATEs [2], FIDES [5], Grøstl [17], LED [20], and Prøst [21] adopt various (n, m)-AES-like primitives other than $(4, 4)$-AES-like primitives, for example.

M. Manulis et al. (Eds.): ACNS 2016, LNCS 9696, pp. 467–484, 2016.
DOI: 10.1007/978-3-319-39555-5_25

Table 1. Lower bounds of 4-round (n, m)-AES-like primitives when $n \leq m$.

Dimension n	Type	Best branch number	Classical bound	Enhanced bound
4	MDS	5	25	-
	Binary	4	16	16 (max)
5	MDS	6	36	-
	Binary	4	16	17 (max)
6	MDS	7	49	-
	Binary	4	16	24 (max)
7	MDS	8	64	-
	Binary	4	16	24[a]
8	MDS	9	81	-
	Binary	5	25	32 (max)

[a] Enhancement is maximized for AES-like primitive with an $(7, 7)$ matrix state.

Recently, many lightweight primitives have been proposed, and they are expected to perform well in area-constrained and low-power environments as well as high-end environments. MixColumns in the original AES adopts a 4×4 Maximum Distance Separable code (MDS) matrix and its elements only take '1', '2', and '3', which is one of the best choices with respect to the cost of multiplication in a Galois field and branch number [30]. However, if the area is very constrained, even the multiplication of an MDS matrix becomes disadvantage for lightweight implementation. There are two methods for reducing the cost of multiplication for both lightweight and high-end environments. One involves a recursive approach [2,19,20] and the other involves a binary matrix similar to Camellia P-function [3,5,31]. In the recursive approach, an MDS matrix is generated by an iterating lightweight matrix, and it is superior to classical MDS matrices for area-constrained lightweight implementation. However, the execution time tends to be slow, which means that it also requires high power consumption because of the recursive operation [15]. On the other hand, the use of a binary matrix is also superior to classical MDS matrices for both constrained and non-constrained environments because it can be implemented only by element-wise XORs[1]. Unfortunately, the branch number of a binary matrix is lower than that of an MDS matrix. For instance, when $\mathcal{B}$ denotes the differential and linear branch number of the matrix, AES-like primitives guarantee at least $\mathcal{B}^2$ active S-boxes in 4-round differential and linear characteristics [10]. Therefore, AES-like primitives with a binary matrix have fewer active S-boxes than those with an MDS matrix, and it requires more rounds to guarantee security against several cryptanalyses.

[1] When a matrix is an $n \times n$ matrix whose elements take ℓ-bit value, both an MDS and a "binary" matrices are also represented by binary matrices on $(\mathbb{F}_2)^{\ell n \times \ell n}$. Then, the Hamming weight of MDS matrix is always greater than ℓn^2, but that of "binary" matrix is smaller than ℓn^2.

Our Contribution. In this paper, we focus on binary matrices and discuss useful cryptographic properties of binary matrices. We specifically focus on AES-like primitives with *binary MixColumns*, whose output is computed using a binary matrix.

If the number of active S-boxes per specific number of rounds increases, we can efficiently guarantee that the block cipher with fewer rounds has immunity against several cryptanalyses. In previous design criteria, we only care about the branch number of binary matrices because the classical proof only guarantees $\mathcal{B}^2$ active S-boxes in the 4-round characteristic. However, we argue that the classical lower bound is not tight for some binary matrices. Namely, there are binary matrices such that the lower bound is more than $\mathcal{B}^2$.

In this paper, we exhaustively search $n \times n$ binary matrices with $n \in \{4, 5, \ldots, 8\}$ and show some instances whose lower bound is more than $\mathcal{B}^2$. We first discuss cryptographic properties of binary matrices. Then, we propose an algorithm to evaluate a more accurate lower bound by using these properties. Our algorithm efficiently evaluates the lower bound for a given binary matrix, and some matrices enhance the lower bound from $\mathcal{B}^2$ to $\mathcal{B}(\mathcal{B}+2)$. Specifically, our algorithm finds some binary matrices whose lower bounds become 16, 17, 24, 24, and 32 for $n = 4, 5, 6, 7$, and 8, respectively. We summarize the enhanced lower bounds in Table 1. Since the highest branch number of binary matrices is 4 for $n \in \{4, 5, \ldots, 7\}$, the classical proof only guarantees 16 active S-boxes. Moreover, since the highest branch number is 5 for $n = 8$, the classical proof only guarantees 25 active S-boxes. Therefore, we can enhance the lower bounds for $n \in \{5, 6, 7, 8\}$. We also evaluate the limit of the enhancement. We guarantee that the enhancement in Table 1 is maximized for all (n, n)-AES-like primitives with $n \in \{4, 5, \ldots, 8\}$. Moreover, for all (n, m)-AES-like primitives with $n < m$, we also guarantee that the enhancement is maximized for $n \in \{4, 5, 6, 8\}$.

2 Preliminaries

2.1 Definitions

Notations. Let $x = (x_1, x_2, \ldots, x_n)$ be an n-dimensional vector over $\mathbb{F}_{2^\ell}$. Let $x[j] = (x_1[j], x_2[j], \ldots, x_n[j])$ be an n-dimensional vector over $\mathbb{F}_2$, where $x_i[j]$ denotes the jth bit in x_i. Let $\tilde{x} \in (\mathbb{F}_2)^n$ be the truncation of $x \in (\mathbb{F}_{2^\ell})^n$ such that the ith element of $\tilde{x}$, i.e., $\tilde{x}_i$ takes 0 if $x_i = 0$ and takes 1 if $x_i \neq 0$. The Hamming weight of $x_i \in \mathbb{F}_{2^\ell}$ is calculated as $hw(x_i) = \sum_{j=1}^{\ell} x_i[j]$, where the addition is calculated over $\mathbb{Z}$. Moreover, the Hamming weight of $x \in (\mathbb{F}_{2^\ell})^n$ is calculated based on the truncated vector, i.e., it is calculated as $hw(x) = \sum_{i=1}^{n} \tilde{x}_i$. For any $a \in \mathbb{F}_2^n$ and $b \in \mathbb{F}_2^n$, let $a \succeq b$ if $a \vee b = a$, where $\vee$ denotes a bit-wise OR. Note that an element in $\mathbb{F}_{2^\ell}$ is represented as an ℓ-bit vector in $\mathbb{F}_2^\ell$, and it is naturally converted using an appropriate basis.

Active S-boxes. When we evaluate security against differential and linear cryptanalyses, we often evaluate the number of active S-boxes. An S-box that

has a non-zero input difference is called a *differentially active S-box*, and an S-box that has a non-zero output linear mask is called a *linearly active S-box*. We can show the "provable security" against the differential and linear cryptanalyses by guaranteeing the lower bound of the number of active S-boxes.

The Substitution Permutation Network (SPN) cipher based on the wide trail design strategy [12] consists of a confusion layer and diffusion layer, where parallel applications of S-boxes and matrix multiplications are used in the confusion layer and diffusion layer, respectively. When ℓ-bit S-boxes are applied in the confusion layer, the diffusion matrix M is represented as $(\mathbb{F}_{2^\ell})^{n\times n}$ matrix. Let $x \in (\mathbb{F}_{2^\ell})^n$ be the input of the diffusion represented by an M. Then, the output is calculated as $y^T = Mx^T$. To evaluate the security of the diffusion matrix, we often focus on the branch number.

Definition 1 (Branch Number [30]). *Let M be an $n \times n$ matrix over $\mathbb{F}_{2^\ell}$. Then, a differential branch number of M is defined as $\mathcal{B}_d = \min\{hw(x) + hw(Mx^T) \mid x \in (\mathbb{F}_{2^\ell})^n \setminus \{0\}\}$. Similarly, a linear branch number of M is defined as $\mathcal{B}_l = \min\{hw(yM) + hw(y) \mid y \in (\mathbb{F}_{2^\ell})^n \setminus \{0\}\}$.*

Note that $\mathcal{B}_d$ and $\mathcal{B}_\ell$ is always less than or equal to $n + 1$. In the following sections, we only consider differential cryptanalysis unless otherwise noted. For linear cryptanalysis, similar discussion can be made because of the duality of these cryptanalyses [27].

We call that two $n\times n$ matrices M and M' are permutation-homomorphic [24] to each other if there is a row permutation ρ and a column permutation γ satisfying $\rho(\gamma(M)) = \gamma(\rho(M)) = M'$.

Lemma 1 [24]**.** *Let M and M' be matrices that are permutation-homomorphic to each other. Then M and M' have the same differential and linear branch number.*

In cryptographic applications, an MDS matrix has good properties and is defined in the context of coding theory. Its definition is equivalent as the following theorem for our context.

Theorem 1 [30]**.** *Let M be an $n \times n$ MDS matrix, the differential and linear branch number is $n + 1$.*

It is very useful to use the MDS matrix in the diffusion layer since the branch number takes the maximum possible value. However, it is inefficient for lightweight implementation because the multiplication by the MDS matrix requires the multiplication in a Galois field. On the other hand, if all elements of the matrix consist of binary elements, we can efficiently implement the multiplication because it only requires ℓ-bitwise XORs. Unfortunately, such a binary matrix does not generate an MDS matrix except for the trivial MDS matrix, i.e., $n = 1$. Nevertheless, there are concrete ciphers that adopt binary matrices. For example, Camellia uses an 8×8 binary matrix [3], and the designers showed that the maximum branch number of 8×8 binary matrices is 5 from computation using a PC. Kwon et al. summarized the maximum branch number of binary matrix with $n = 4, 5, 6, 7$, and 8 as $4, 4, 4, 4$, and 5, respectively, and they call such matrices *Maximum Distance Binary Linear (MDBL) matrices* [25].

2.2 AES-Like Primitives

The state of AES is represented as a 4×4 matrix whose elements take 8-bit values, i.e., the block length is 128 bits. Many cryptographic primitives use similar state expressions, and we call them AES-like primitives [2,5,10,17,19–21].

We only focus on the property of AES-like primitives independent of a choice of S-boxes. For convenience, let ℓ be the bit length of the input and output of an S-box. We introduce (n, m)-AES-like primitives, where the numbers of rows and columns are scaled like [8].

Definition 2 ((n, m)-AES-Like Primitives). *The AES-like primitives are parameterized by n and m, where the state is represented as an $n \times m$ matrix and $m \geq n$. The round function consists of four component functions: SubBytes, ShiftRows, MixColumns, and AddRoundKey. Each function is defined as follows:*

- *SubBytes (SB) substitutes each ℓ-bit value in the matrix into another ℓ-bit value by an S-box.*
- *ShiftRows (SR) rotates each ℓ-bit value located at row i by i positions to the left.*
- *MixColumns (MC) diffuses n ℓ-bit values within each column by a linear function.*
- *AddRoundKey (AK) XORs the round key with the state.*

Then, the round function of an AES-like primitive is defined as

$$Y \leftarrow (MC \circ SR \circ SB)(X) \oplus RK,$$

where X, Y, and RK denote the input, output, and round key, respectively. When a cryptographic permutation is designed, a constant is XORed to the matrix state instead of a round key.

We also focus on the following MixColumns.

Definition 3 (Binary MixColumns). *When the AES-like primitive uses a binary matrix in the MixColumns, we call such MixColumns* binary MixColumns.

Figure 1 shows 4-round AES-like primitives, which are equivalently transformed with regard to counting the number of active S-boxes. When analyzing 4-round AES-like primitives, we divide the primitive into three layers; front, middle, and back, as shown in Fig. 1. We often focus on the so-called *super-S-box* [13,18], which is defined as follows.

Definition 4 (Super-S-box). *Let a super-S-box consist of two S-box layers and one MixColumns. First, n S-boxes are applied. Then, a diffusion matrix M is applied. Finally, n S-boxes are applied again.*

If the branch number of M is $\mathcal{B}$, an active super-S-box has at least $\mathcal{B}$ active S-boxes. Moreover, both the front and the back layers of the AES-like primitives have m super-S-boxes, respectively.

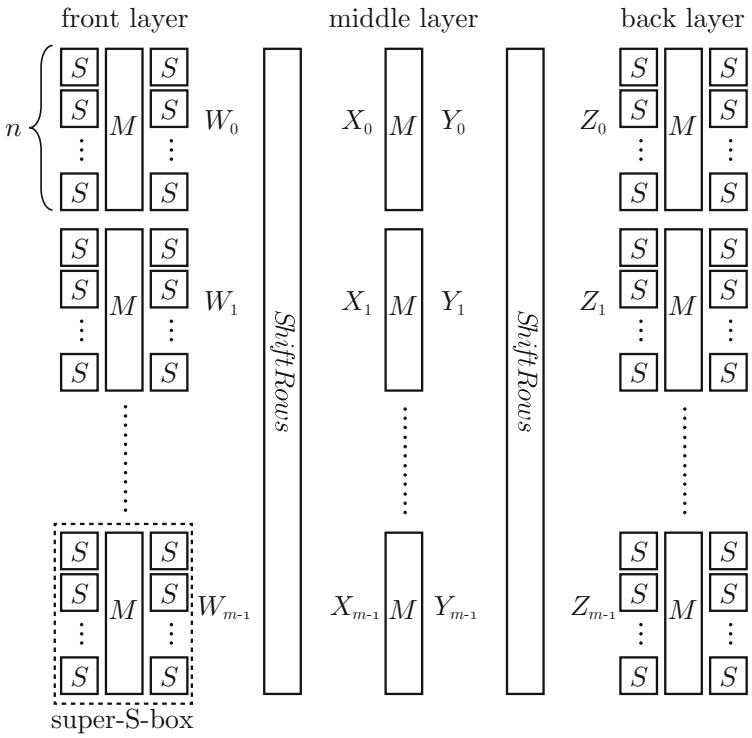

Fig. 1. Proof for 4-round AES-like primitives

Number of Active S-boxes. A good property of AES-like primitives is that the number of active S-boxes in the 4-round characteristic independent of a choice of S-boxes and AddRoundKey can be guaranteed[2]. First, all (n, m)-AES-like primitives have the following characteristic.

Lemma 2. *Let M be an $n \times n$ matrix over $\mathbb{F}_{2^\ell}$. Let $\mathcal{B}$ be the branch number of M. When M is adopted in MixColumns of AES-like primitives, there is always a 4-round characteristic whose number of active S-boxes is lower than or equal to $(n+1)\mathcal{B}$ active S-boxes.*

Proof. Let us focus on the middle layer in Fig. 1. Since the branch number of M is $\mathcal{B}$, there is always a 4-round characteristic satisfying $hw(X_0) + hw(Y_0) = \mathcal{B}$. Then, $hw(X_0) + hw(Y_0)$ super-S-boxes are active, and each super-S-box has at most $n + 1$ active S-boxes. Therefore, there is always a 4-round characteristic whose number of active S-boxes has at most

$$(n+1)hw(X_0) + (n+1)hw(Y_0) = (n+1)(hw(X_0) + hw(Y_0)) = (n+1)\mathcal{B}.$$

□

[2] Any part of this paper does not consider the trivial characteristic that has no active S-box.

Next, let us consider the lower bound of the number of active S-boxes.

Lemma 3 [11]. *Let M be an $n \times n$ matrix over $\mathbb{F}_{2^\ell}$. Let $\mathcal{B}$ be the branch number of M. When M is applied to the MixColumns in AES-like primitives, there are at least $\mathcal{B}^2$ active S-boxes in the 4-round characteristic.*

Lemmas 2 and 3 derive the following theorem.

Theorem 2. *Assuming that M is an MDS matrix with branch number $\mathcal{B}$, there are at least $\mathcal{B}^2$ active S-boxes in the 4-round characteristic, and it is tight.*

Theorem 2 shows that there is no MDS matrix in which the minimum number of active S-boxes is more than $\mathcal{B}^2$ in the 4-round characteristic. However, if binary MixColumns is used, there is a possibility that the minimum number of active S-boxes is more than $\mathcal{B}^2$ because $\mathcal{B}^2 < (n+1)\mathcal{B}$. For instance, if a 5×5 binary matrix is used, $\mathcal{B}^2 = 16$ and $(n+1)\mathcal{B} = 24$, and there is a possibility that the minimum number of active S-boxes can be improved to 24.

3 Properties of Binary Matrices

We now discuss useful properties of binary matrices. Let $x \in (\mathbb{F}_{2^\ell})^n \setminus \{0\}$ be the input difference. Specifically, we focus on the propagation $x \xrightarrow{M} Mx^T$. Assume that the branch number of M is $\mathcal{B}$, i.e., $hw(\tilde{x}) + hw(\widetilde{Mx^T})$ is at least $\mathcal{B}$. Then, an enhanced propagation is defined as follows.

Definition 5 (Enhanced Propagation). *For a binary matrix $M \in (\mathbb{F}_{2^\ell})^{n \times n}$ with branch number $\mathcal{B}$, $x \in (\mathbb{F}_{2^\ell})^n \setminus \{0\}$ denotes the input difference of the diffusion by M. We say that the propagation $x \xrightarrow{M} Mx^T$ is an enhanced propagation, when $hw(\tilde{x}) + hw(\widetilde{Mx^T}) > \mathcal{B}$.*

When we consider all possible propagations from x, the minimum of $hw(\tilde{x}) + hw(\widetilde{Mx^T})$ is $\mathcal{B}$ because of the branch number. However, some propagations have $hw(\tilde{x}) + hw(\widetilde{Mx^T}) > \mathcal{B}$. Moreover, we define the following two propagations.

Definition 6 (Direct and Indirect Propagations). *For a binary matrix $M \in (\mathbb{F}_{2^\ell})^{n \times n}$, $x \in (\mathbb{F}_{2^\ell})^n \setminus \{0\}$ denotes the input difference of the diffusion by M. We say that the propagation $x \xrightarrow{M} Mx^T$ is a direct (resp. indirect) propagation, when $\widetilde{Mx^T} = M\tilde{x}^T$ (resp. $\widetilde{Mx^T} \neq M\tilde{x}^T$).*

In the direct propagation, $\widetilde{Mx^T}$ can be directly calculated from $\tilde{x}$ as $M\tilde{x}^T$. In the indirect propagation, we cannot calculate $\widetilde{Mx^T}$ from only $\tilde{x}$ and have to calculate it from the difference x.

3.1 Indirect Branch Number

We now want to evaluate the propagation $x \xrightarrow{M} Mx^T$, and let us consider the condition in which the propagation becomes an enhanced propagation. We first define a variant of the branch number as follows.

Definition 7 (Indirect Branch Number). *Let M be an $n \times n$ binary matrix over $\mathbb{F}_{2^\ell}$. Let $x \in (\mathbb{F}_{2^\ell})^n \setminus \{0\}$ be the input difference of the diffusion by M. For all indirect propagations, i.e., all $x \xrightarrow{M} Mx^T$ satisfying $\widetilde{Mx^T} \neq M\tilde{x}^T$, the indirect branch number denotes the minimum of $hw(\tilde{x}) + hw(\widetilde{Mx^T})$.*

We can obtain a useful lemma about the indirect branch number.

Lemma 4. *Let M be an $n \times n$ binary matrix over $\mathbb{F}_{2^\ell}$. Let $\mathcal{B}$ be the branch number of M, and assume $\mathcal{B} > 2$. Then, the indirect branch number is at least $\mathcal{B} + 2$.*

Proof. Let y be the output vector, i.e., $y^T = Mx^T$. When the propagation $x \xrightarrow{M} y$ is indirect propagation, i.e., $\tilde{y}^T \neq M\tilde{x}^T$, there are always two non-zero $x[i]$ and $x[j]$ satisfying $x[i] \neq x[j]$, and $hw(\tilde{x}) \geq hw(x[i] \vee x[j])$. Similarly, let $y[i]^T = Mx[i]^T$ and $y[j]^T = Mx[j]^T$, and $hw(\tilde{y}) \geq hw(y[i] \vee y[j])$. Without loss of generality, assume $hw(x[j]) + hw(y[j]) \geq hw(x[i]) + hw(y[i])$.

First, assuming that $hw(x[j]) + hw(y[j]) \geq \mathcal{B} + 2$, the sum of the Hamming weight of $\tilde{x}$ and that of $\tilde{y}$ is at least $\mathcal{B} + 2$.

Second, assume that $hw(x[j]) + hw(y[j]) = \mathcal{B} + 1$. When $x[j] \not\succeq x[i]$, $hw(x[i] \vee x[j]) \geq hw(x[j]) + 1$. Moreover, when $y[j] \not\succeq y[i]$, $hw(y[i] \vee y[j]) \geq hw(y[j]) + 1$. Therefore, when $x[j] \not\succeq x[i]$ or $y[j] \not\succeq y[i]$, the sum of the Hamming weight of $\tilde{x}$ and that of $\tilde{y}$ is at least $\mathcal{B} + 2$ because

$$hw(x[i] \vee x[j]) + hw(y[i] \vee y[j]) \geq hw(x[j]) + hw(y[j]) + 1 = \mathcal{B} + 2.$$

Finally, when $x[j] \succeq x[i]$ and $y[j] \succeq y[i]$,

$$\begin{aligned} hw(x[i] \oplus x[j]) + hw(y[i] \oplus y[j]) &= hw(x[j]) - hw(x[i]) + hw(y[j]) - hw(y[i]) \\ &\leq \mathcal{B} + 1 - \mathcal{B} = 1, \end{aligned}$$

where $(y[i] \oplus y[j])^T = M(x[i] \oplus x[j])^T$. Therefore, this is contradictory because the branch number is greater than 2.

Third, assuming that $hw(x[j]) + hw(y[j]) = \mathcal{B}$, $hw(x[i]) + hw(y[i]) = \mathcal{B}$. Without loss of generality, assume $hw(x[j]) \geq hw(x[i])$. When $hw(x[i]) = hw(x[j])$, $hw(x[i] \vee x[j]) \geq hw(x[j]) + 1$ because $x[i] \neq x[j]$. Moreover, $hw(y[i] \vee y[j]) \geq hw(y[j]) + 1$ because $y[i] \neq y[j]$. Therefore, the sum of the Hamming weight of $\tilde{x}$ and that of $\tilde{y}$ is at least $\mathcal{B} + 2$ because

$$hw(x[i] \vee x[j]) + hw(y[i] \vee y[j]) \geq hw(x[j]) + 1 + hw(y[j]) + 1 = \mathcal{B} + 2.$$

When $hw(x[i]) + 1 = hw(x[j])$, then $hw(y[i]) = hw(y[j]) + 1$. If $x[j] \not\succeq x[i]$, $hw(x[i] \vee x[j]) \geq hw(x[j]) + 1 = hw(x[i]) + 2$. Moreover, if $y[i] \not\succeq y[j]$, $hw(y[i] \vee$

$y[j]) \geq hw(y[i]) + 1 = hw(y[j]) + 2$. Therefore, when $x[j] \not\succeq x[i]$ or $y[i] \not\succeq y[j]$, the sum of the Hamming weight of $\tilde{x}$ and that of $\tilde{y}$ is at least $\mathcal{B} + 2$. Finally, when $x[j] \succeq x[i]$ and $y[i] \succeq y[j]$,

$$\begin{aligned} hw(x[i] \oplus x[j]) + hw(y[i] \oplus y[j]) &= hw(x[j]) - hw(x[i]) + hw(y[i]) - hw(y[j]) \\ &= 1 + 1 = 2. \end{aligned}$$

Therefore, this is contradictory because the branch number is greater than 2. When $hw(x[i]) + 2 \leq hw(x[j])$, then the sum of the Hamming weight of $\tilde{x}$ and that of $\tilde{y}$ is at least $\mathcal{B} + 2$ because

$$hw(x[i] \vee x[j]) + hw(y[i] \vee y[j]) \geq hw(x[i]) + 2 + hw(y[i]) = \mathcal{B} + 2.$$

□

Lemma 4 shows that the indirect propagation is always an enhanced propagation when $\mathcal{B} > 2$.

3.2 Propagation on Restricted Input and Output Differences

When we consider the propagation $x \xrightarrow{M} Mx^T$, $hw(\tilde{x}) + hw(\widetilde{Mx^T})$ is generally lower-bounded by branch number. However, if Hamming weight of input difference or that of output difference is restricted, it is not always lower-bounded by the branch number, i.e., it may have higher lower bounds.

Lemma 5. *Let M be an $n \times n$ binary matrix over $\mathbb{F}_{2^\ell}$. Let $\mathcal{B}$ be the branch number. Let $x \in (\mathbb{F}_{2^\ell})^n \setminus \{0\}$ be the input difference of the diffusion by M. Then, assuming that $hw(\tilde{x}) \leq 2$,*

$$hw(\tilde{x}) + hw(\widetilde{Mx^T}) \geq hw(\tilde{x}) + hw(M\tilde{x}^T).$$

Similarly, assuming that $hw(\widetilde{Mx^T}) \leq 2$,

$$hw(\tilde{x}) + hw(\widetilde{Mx^T}) \geq hw(M^{-1}(\widetilde{Mx})^T) + hw(\widetilde{Mx^T}).$$

Proof. We prove the first part of the lemma. Both left- and right-hand sides of the inequality include the term $hw(\tilde{x})$; thus, it is sufficient to prove $hw(\widetilde{Mx^T}) \geq hw(M\tilde{x}^T)$. Both $\widetilde{Mx^T}$ and $M\tilde{x}^T$ can be regarded as a truncated difference, so we focus on these truncated differences. For the right-hand side, $M\tilde{x}^T$, only $\mathbb{F}_2$-operations are performed. For the left-hand side, $\widetilde{Mx^T}$, we need to consider the following steps; 1. convert the truncated difference to (full) difference, 2. multiply matrix M, and 3. reconvert the difference to truncated difference. Therefore, we need to consider the following "special" operation for truncated differences 0 and 1: $0 \oplus 0 = 0$, $0 \oplus 1 = 1$, $1 \oplus 0 = 1$, and $1 \oplus 1 = 0$ or 1. Recall that we are evaluating Hamming weight. Thus, when $1 \oplus 1 = 1$, the left-hand side is greater than the right-hand side; otherwise they are equal. The second part of the lemma can be obtained to substitute x and M with Mx^T and M^{-1}, respectively. □

Assuming that the Hamming weight of the input difference or that of the output difference is at most 2, Lemma 5 shows that $hw(\tilde{x}) + hw(\widetilde{Mx^T})$ can be lower-bounded by the corresponding direct propagation. Therefore, we can effectively guarantee the lower bound of $hw(\tilde{x}) + hw(\widetilde{Mx^T})$. Specifically, let us consider the time complexity to guarantee the lower bound. Then, the time complexity is $O(n)$ when the Hamming weight is at most 1, and it is $O(n(n-1))$ when the Hamming weight is at most 2.

4 Number of Active S-boxes in AES-Like Primitives with Binary MixColumns

From Lemma 2, there is always a 4-round characteristic whose number of active S-boxes is lower than or equal to $(n+1)\mathcal{B}$, and the use of MDS matrices is the best choice because $\mathcal{B}^2 = \mathcal{B}(n+1)$. However, if a binary MixColumns is used, there is a gap between $\mathcal{B}^2$ and $\mathcal{B}(n+1)$ since $\mathcal{B} < n+1$. In this section, we guarantee more accurate lower bound of the number of active S-boxes in the 4-round characteristic. Note that our proof is independent of the choice of S-boxes.

4.1 Intuition of Idea

First, we revisit the proof that there are at least $\mathcal{B}^2$ differentially and linearly active S-boxes in the 4-round characteristic of the AES-like primitives. We focus on the propagation in the middle layer, and we assume that the ith MixColumns is active. Then $hw(\tilde{x}) + hw(\widetilde{Mx^T})$ is at least $\mathcal{B}$, and there are at least $\mathcal{B}$ active super-S-boxes in the 4-round characteristic because of the property of SR. Since every active super-S-box has $\mathcal{B}$ active S-boxes, there are at least $\mathcal{B}^2$ active S-boxes in the 4-round characteristic.

Now, we consider an AES-like primitive whose MixColumns uses a binary matrix with branch number $\mathcal{B}$.

First, we consider the case in which there is an indirect propagation in the middle layer. Since the indirect branch number is $\mathcal{B}+2$ from Lemma 4, there are at least $\mathcal{B}+2$ active super-S-boxes in the 4-round characteristic. This also implies that there are at least $\mathcal{B}(\mathcal{B}+2)$ active S-boxes in the 4-round characteristic.

Next, we consider the case in which there is an only direct propagation in the middle layer. We focus on the number of active MixColumns in the middle layer, and i active MixColumns denote the case in which i MixColumns are active in the middle layer. Then, the minimum number of active S-boxes is proven using different methods depending on the number of active MixColumns. In more detail, let us consider the following cases, where the notation in Fig. 1 is used, and Fig. 2 shows the outline. First, we assume i active MixColumns with $i \leq 2$. Then, at most two elements in W_i and Z_i are active for any i because of the construction of SR. Therefore, we effectively guarantee the minimum number of active S-boxes in every super-S-box using Lemma 5. Next, we assume i active

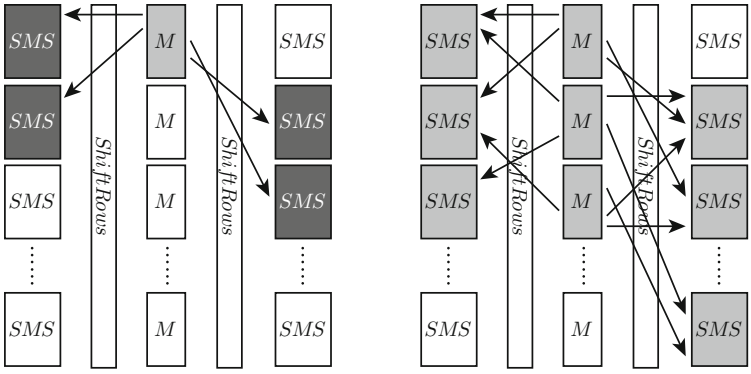

Fig. 2. Proof Strategy. When the number of active MixColumns is at most two (see the left figure), we use a binary matrix M such that super-S-boxes in the front and back layers always have enhanced propagation. When the number of active MixColumns is at least three (see the right figure), we use an M such that the characteristics always have many active super-S-boxes.

MixColumns with $i \geq 3$. We choose binary matrices such that the number of active super-S-boxes is beyond $\mathcal{B}$ for all characteristics.

Section 4.2 shows an algorithm to efficiently evaluate a more accurate lower bound of a given binary matrix.

4.2 Algorithm to Obtain Accurate Lower Bound

We guarantee the lower bound for a given binary matrix $M \in \mathbb{F}_{2^\ell}^{n \times n}$, and Algorithm 1, the validity of which is shown later in this section, shows the procedure to evaluate a more accurate lower bound. Here, AS_i and ASS_i are defined as follows.

Definition 8 (AS_i : Accurate lower bound of number of active S-boxes under i active MixColumns on direct propagation). *We only consider the 4-round characteristic whose propagation does not have the indirect propagation. For any characteristic with i active MixColumns in the middle layer, AS_i denotes the accurate lower bound of the number of active S-boxes in the 4-round characteristic.*

Definition 9 (ASS_i : Accurate lower bound of number of active super-S-boxes under i active MixColumns on direct propagation). *We only consider the 4-round characteristic whose propagation does not have the indirect propagation in the middle layer. For any characteristic with i active MixColumns in the middle layer, ASS_i denotes the accurate lower bound of the number of active super-S-boxes in the 4-round characteristic.*

Both AS_i and ASS_i only focus on characteristics whose middle layer has direct propagations. Moreover, AS_i only focuses on the characteristic whose super-S-boxes have direct propagations, but the bound $\mathcal{B} \times ASS_i$ takes into account

Algorithm 1. Algorithm to obtain accurate lower bound

Input: A binary matrix $M \in \mathbb{F}_{2^\ell}^{n \times n}$.
Output: The lower bound of the number of active S-boxes in the 4-round characteristic.

procedure `AccurateBound`(M)
  Calculate $\mathcal{B}$ as the branch number of M.
  Calculate AS_1 and ASS_2. ▷ See Definitions 8 and 9.
  if $AS_1 \leq \min\{\mathcal{B} \times ASS_2, \mathcal{B}(\mathcal{B}+2)\}$ **then**
    return AS_1
  else
    Calculate AS_2 and ASS_3.
    if $\min\{AS_1, AS_2\} \leq \min\{\mathcal{B} \times ASS_3, \mathcal{B}(\mathcal{B}+2)\}$ **then**
      return $\min\{AS_1, AS_2\}$
    else
      return $\min\{\mathcal{B} \times ASS_3, \mathcal{B}(\mathcal{B}+2)\}$
    end if
  end if
end procedure

characteristics whose super-S-boxes have indirect propagations. Therefore, $\mathcal{B} \times ASS_i \leq AS_i$. Moreover, ASS_i monotonically increases as a value of i.

For any binary matrix M with branch number $\mathcal{B}$, the number of active S-boxes in the 4-round characteristic is lower-bounded by

$$\min\{\mathcal{B} \times ASS_1, \mathcal{B}(\mathcal{B}+2)\}. \tag{1}$$

Here, $\mathcal{B} \times ASS_1$ and $\mathcal{B}(\mathcal{B}+2)$ denote the lower bound in which the middle layer has an only direct propagation and indirect propagation, respectively. Note that since $ASS_1 = \mathcal{B}$, the number of active S-boxes is lower-bounded by $\mathcal{B} \times ASS_1 = \mathcal{B}^2$.

We first calculate AS_1 to obtain a more accurate lower bound. Since AS_1 only focuses on the characteristic whose propagations do not have indirect propagations and there is at most one active MixColumns, it can be computed by counting the number of Hamming weights of the column vector of M and M^{-1} by considering the computation of the multiplication by M and M^{-1}.

$$AS_1 = \min_{\tilde{x} \in \mathbb{F}_2^n \setminus \{0\}} \left\{ \sum_{i=1}^{n} (hw((M^{-1})_i)\tilde{x}_i + hw(M_i)(M\tilde{x}^T)_i) \right\},$$

Note that M_i and $(M^{-1})_i$ denote the ith column vector in M and M^{-1}, respectively, and AS_1 does not depend on the position of the active MixColumns in the middle layer. Therefore, we can obtain AS_1 with $O(2^n)$ time complexity. Since Lemma 5 enables us only to consider the case of direct propagations, we can replace $\mathcal{B} \times ASS_1$ with $\min\{AS_1, \mathcal{B} \times ASS_2\}$ in (1). Then, the number of active S-boxes is lower-bounded by

$$\min\{AS_1, \mathcal{B} \times ASS_2, \mathcal{B}(\mathcal{B}+2)\}. \tag{2}$$

Note that there is always a characteristic whose number of active S-boxes is AS_1. Therefore, AS_1 is a tight lower bound if $AS_1 \leq \min\{\mathcal{B} \times ASS_2, \mathcal{B}(\mathcal{B}+2)\}$. Otherwise, $\min\{\mathcal{B} \times ASS_2, \mathcal{B}(\mathcal{B}+2)\}$ is a new lower bound, but we do not guarantee whether or not it is tight.

When $AS_1 > \mathcal{B} \times ASS_2$, there is a possibility that the lower bound can be further improved. Lemma 5 shows that we can replace $\mathcal{B} \times ASS_2$ with $\min\{AS_2, \mathcal{B} \times ASS_3\}$ in (2). Then, the number of active S-boxes is lower-bounded by

$$\min\{AS_1, AS_2, \mathcal{B} \times ASS_3, \mathcal{B}(\mathcal{B}+2)\}. \tag{3}$$

Since both AS_2 and ASS_2 depend on truncated differentials of two active MixColumns and the difference between positions of two active MixColumns, we can obtain them with $O((n-1) \times 2^{2n})$ time complexity. Similarly, since ASS_3 depends on truncated differentials of three active MixColumns and the difference among positions of three active MixColumns, we can obtain it with $O((n-1)(n-2) \times 2^{3n})$ time complexity. Note that there are always characteristics whose number of active S-boxes is AS_2. Therefore, $\min\{AS_1, AS_2\}$ is a tight lower bound if $\min\{AS_1, AS_2\} \leq \min\{\mathcal{B} \times ASS_3, \mathcal{B}(\mathcal{B}+2)\}$. Otherwise, $\min\{\mathcal{B} \times ASS_3, \mathcal{B}(\mathcal{B}+2)\}$ is a new lower bound, but we cannot guarantee whether or not it is tight. Note that tightness is not efficiently guaranteed because we cannot use Lemma 5 for three active MixColumns.

For linear cryptanalysis, we also execute the same procedure for the binary matrix M^T because of the duality between differential and linear cryptanalyses (see Appendix A).

5 Best Binary Matrices

We now want to evaluate all $n \times n$ binary matrices and efficiently obtain binary matrices whose number of active S-boxes is maximized in the 4-round characteristic.

5.1 Efficient Search

The number of $n \times n$ binary matrices is 2^{n^2}, and e.g., since 2^{64} for $n = 8$, it is infeasible to exhaustively evaluate all matrices. However, in the application to MixColumns, we usually prefer to use binary matrices with the highest branch number. Therefore, we exhaustively search binary matrices with the highest branch number from $n = 4$ to $n = 8$ by using a similar technique to that by Guo et al. [16].

Fact 1. *For binary matrices with* $n = 4, 5, 6, 7,$ *and* 8*, the numbers of binary matrices with the highest differential and linear branch number are* $4! \approx 2^{4.6}$*,* $22 \times 5! \approx 2^{11.4}$*,* $49032 \times 6! \approx 2^{25.1}$*,* $279631988 \times 7! \approx 2^{40.4}$*, and* $18527040 \times 8! \approx 2^{39.4}$*, respectively.*

Moreover, we only consider invertible binary matrices.

Algorithm 1 requires much time complexity. Note that there is always a characteristic whose number of active S-boxes is equal to AS_1. Then, the lower bound of the number of active S-boxes is always upper-bounded by at most AS_1. Therefore, we first exhaustively search all binary matrices with the highest branch number and only evaluate AS_1. Table 2 shows AS_1, where columns in DC and those in LC have AS_1 of M and that of M^T, respectively. Columns in DC corresponds to the case for differential characteristics and columns in LC corresponds to the case for linear characteristics. Moreover, Table 2 does not include the case in which AS_1 for DC is greater than that for LC. When the number of columns is greater than or equal to the number of rows, we can calculate AS_1 independent of the number of columns. Therefore, from Table 2, we obtain the following fact.

Table 2. AS_1 of all $MDBL$ matrices with $n = 4, 5, \ldots, 8$.

n	DC	LC	# of matrices
4	16	16	24
5	16	16	2160
5	17	17	480
6	16	16	5650560
6	16	17	4364640
6	16	18	1011600
6	16	19	15840
6	16	20	2160
6	17	17	9405360
6	17	18	2821680
6	17	19	90720
6	18	18	2586240
6	18	19	244800
6	18	20	27360
6	19	19	275040
6	19	20	54720
6	20	20	103680
6	21	21	11520
6	22	22	2880
6	24	24	720
7	16	16	22453467120
7	16	17	43355400480
7	16	18	34791593760
7	16	19	9488802960
7	16	20	1606162320
7	16	21	70817040
7	16	22	2716560
7	16	24	90720
7	17	17	126753399360
7	17	18	132789625920
7	17	19	49796596080
7	17	20	10055893680
7	17	21	640024560
7	17	22	27649440
7	17	23	70560
7	18	18	200729783520
7	18	19	105763669200
7	18	20	29003380560
7	18	21	2736417600
7	18	22	160644960
7	18	23	1547280
7	18	24	594720
7	19	19	88863979680
7	19	20	36434255760
7	19	21	5529872880
7	19	22	483537600
7	19	23	9051840
7	19	24	1149120
7	20	20	24798715200
7	20	21	6400180080
7	20	22	923988240
7	20	23	33405120
7	20	24	3417120
7	21	21	3160795680
7	21	22	795795840
7	21	23	60490080
7	21	24	4929120
7	21	25	10080
7	22	22	445440240
7	22	23	64506960
7	22	24	9671760
7	22	25	50400
7	23	24	6325200
7	23	25	161280
7	24	24	4969440
7	24	25	30240
7	25	25	40320
8	25	25	126252403200
8	25	26	99931668480
8	25	27	9902471040
8	25	28	214462080
8	25	29	1290240
8	26	26	191120630400
8	26	27	58113216000
8	26	28	3361276800
8	26	29	38868480
8	27	27	53379285120
8	27	28	9583176960
8	27	29	503193600
8	27	30	1612800
8	28	28	7646042880
8	28	29	1739808000
8	28	30	16450560
8	29	29	1305642240
8	29	30	37900800
8	30	30	109992960
8	30	31	33546240
8	31	31	229985280
8	31	32	1290240
8	32	32	5806080

DC: # of differentially active S-boxes, LC: # of linearly active S-boxes

Fact 2. *For all 4-round* (n, m)*-AES-like primitives,* AS_1 *is upper-bounded by* 16*,* 17*,* 24*,* 25*, and* 32 *for* $n = 4, 5, 6, 7$*, and* 8*, respectively.*

Therefore, there are not exist binary matrices such that the lower bound is 17, 18, 25, 26, and 33 for $n = 4, 5, 6, 7$, and 8, respectively.

Finally, we exhaustively search all $n \times n$ binary matrices. First, we evaluate AS_1, and if AS_1 is not maximum possible, we prune the matrix. Then, we evaluate the accurate lower bound by using Algorithm 1. If we can find a binary matrix whose lower bound is the same as AS_1, it is one of the best binary matrices. On the other hand, if we cannot find such a matrix, we also evaluate binary matrices whose AS_1 is not maximum possible by using Algorithm 1.

5.2 Examples

Table 3 shows each example of binary matrices with an enhanced lower bound.

When $n = 4$, there is no binary matrix such that the lower bound of the number of active S-boxes is enhanced. On the other hand, for $n > 4$, we find such matrices. Specifically, when $n = 5, 6$, and 8, the enhancement is maximized because of Fact 2. When $n = 7$, we cannot obtain binary matrices such that the number of active S-boxes is lower-bounded by 25. However, for $m = n$, we also exhaustively evaluate the lower bound of AS_2 and AS_3 because there is always a characteristic whose number of active S-boxes is AS_2 or AS_3. As a result, since there is no binary matrix such that the number of active S-boxes is lower-bounded by 25, the enhancement is maximized. For $(7, m)$-AES-like primitives with $7 < m$, it may be possible that the number of active S-boxes is lower-bounded by 25. However, since Lemma 4 only guarantees $4 \times 6 = 24$ active S-boxes, we have to consider the indirect propagation in the middle layer if we guarantee that the number of active S-boxes is lower-bounded by 25.

5.3 Future Work

Essentially, binary matrices with enhanced lower bound tends to have high Hamming weight. For the lightweight implementation, it is important to consider binary matrices that we can compute the multiplication with low XOR count. We have to consider good trade-off.

Our algorithm deeply utilizes the structure of an AES-like primitive and its properties, and this accelerates the algorithm to compute the bounds and derives good matrices. On the other hand, our algorithm is customized for 4-round AES-like primitives, and the mixed-integer linear programming approach [28] seems useful for more round primitives.

We focused on the number of active S-boxes, which implies "provable security" [22] against differential and linear cryptanalyses. Towards the ultimate security against differential and linear cryptanalysis, there is a long way to evaluate our construction. Differential [26], linear hull [29], and plateau characteristics [14] are the topic of this area. Moreover, a "good" cipher should have a similar security level for each cryptanalysis. Therefore, the next problem we

Table 3. Examples of binary matrices with enhanced lower bound.

Binary matrix	4×4	5×5	6×6	7×7	8×8
Example	$\begin{bmatrix} 0&1&1&1 \\ 1&0&1&1 \\ 1&1&0&1 \\ 1&1&1&0 \end{bmatrix}$	$\begin{bmatrix} 1&1&1&0&0 \\ 1&0&1&1&0 \\ 1&1&0&1&1 \\ 1&0&1&0&1 \\ 0&1&1&1&1 \end{bmatrix}$	$\begin{bmatrix} 1&1&1&1&1&0 \\ 1&1&1&1&0&1 \\ 1&1&0&1&1&1 \\ 0&1&1&1&1&1 \\ 1&1&1&0&1&1 \\ 1&0&1&1&1&1 \end{bmatrix}$	$\begin{bmatrix} 1&1&1&1&0&0&0 \\ 1&1&0&1&0&1&0 \\ 1&0&1&0&1&1&1 \\ 1&1&0&1&1&0&0 \\ 0&0&1&1&1&1&1 \\ 0&1&1&0&1&1&1 \\ 1&1&0&1&0&0&1 \end{bmatrix}$	$\begin{bmatrix} 1&1&1&1&1&0&0&0 \\ 1&1&1&0&0&1&1&0 \\ 1&0&1&1&1&1&1&0 \\ 1&1&1&1&0&0&1&1 \\ 0&1&0&1&1&0&1&1 \\ 1&1&0&1&1&1&0&1 \\ 1&0&0&0&1&1&1&1 \\ 0&1&1&0&1&1&1&1 \end{bmatrix}$
Lower bound	16	17	24	24	32

need to analyze is to confirm security against other cryptanalyses, e.g., impossible differential [4], integral [23], and zero-correlation cryptanalyses [6].

6 Conclusion

We investigated the number of active S-boxes in differential and linear characteristics for 4-round AES-like primitive with binary MixColumns. The number is lower-bounded by $\mathcal{B}^2$ when the branch number of the binary MixColumns is $\mathcal{B}$. However, we showed that the lower bound is not always tight for AES-like primitives with binary MixColumns. To analyze the bound, we first introduced *enhanced propagation* and *(in)direct propagations*, and showed useful properties of binary matrix. Then, we showed how to evaluate an accurate lower bound for a given binary matrix. As a result, we showed that some binary matrices enhance the lower bound from $\mathcal{B}^2$ to $\mathcal{B}(\mathcal{B}+2)$. Specifically, for (n,m)-AES-like primitives with $n = 5, 6, 7$, and 8, we find binary matrices whose lower bound is 17, 24, 24, and 32, respectively. Moreover, we also evaluated the limit of the enhancement, and the enhancement is maximized for all (n,n)-AES-like primitives with $n \in \{4, 5, \ldots, 8\}$. Moreover, for all (n,m)-AES-like primitives with $n < m$, we also guarantee that the enhancement is maximized for $n \in \{4, 5, 6, 8\}$.

A Duality Between Differences and Linear Masks

The duality between differential and linear cryptanalyses was pointed out, and several meanings of duality are known [7,27]. When constructing a differential characteristic, we should know the differential propagation rule for XOR and branch operation. That is, $\Delta z = \Delta x \oplus \Delta y$, where $z \leftarrow x \oplus y$, and $\Delta x = \Delta y = \Delta z$, where $x \leftarrow z$ and $y \leftarrow z$. For linear cryptanalysis, we have $\Gamma x = \Gamma y = \Gamma z$, where $z \leftarrow x \oplus y$, and $\Gamma x \oplus \Gamma y = \Gamma z$, where $x \leftarrow z$ and $y \leftarrow z$ [9,27]. We generalize this propagation rule to any linear transformation.

Let $M \in \mathbb{F}_2^{n \times n}$ be a binary matrix, and let $x \in \mathbb{F}_2^n$ be the input of the diffusion represented by an M. Then, let $y \in \mathbb{F}_2^n$ be the output of the diffusion

represented by M as $y^T = Mx^T$. For the differential propagation, $(\Delta y)^T = M(\Delta x)^T$ trivially holds. For the linear mask propagation, we want to know the linear mask Γx and $\Gamma y \in \mathbb{F}_2^n$ such that $\Gamma y \bullet y = \Gamma x \bullet x$ with probability 1. Using the matrix multiplication, the equation can be written as $\Gamma y y^T = \Gamma x x^T$. That is, $\Gamma y(Mx^T) = (\Gamma y M)x^T = \Gamma x x^T$. Thus, $\Gamma y M = \Gamma x \Leftrightarrow M^T(\Gamma y)^T = (\Gamma x)^T$ should hold and is the propagation rule for the linear mask.

References

1. Specification for the Advanced Encryption Standard (AES): U.S. Department of Commerce/National Institute of Standards and Technology, Federal Information Processing Standards Publication 197 (2001)
2. Andreeva, E., Bilgin, B.B., Bogdanov, A., Luykx, A., Mendel, F., Mennink, B., Mouha, N., Wang, Q., Yasuda, K.: PRIMATEs. CAESAR Proposal (2014). http://primates.ae/
3. Aoki, K., Ichikawa, T., Kanda, M., Matsui, M., Moriai, S., Nakajima, J., Tokita, T.: *Camellia*: a 128-bit block cipher suitable for multiple platforms - design and analysis. In: Stinson, D.R., Tavares, S. (eds.) SAC 2000. LNCS, vol. 2012, pp. 39–56. Springer, Heidelberg (2001)
4. Biham, E., Biryukov, A., Shamir, A.: Cryptanalysis of Skipjack reduced to 31 rounds using impossible differentials. In: Stern, J. (ed.) EUROCRYPT 1999. LNCS, vol. 1592, pp. 12–23. Springer, Heidelberg (1999)
5. Bilgin, B., Bogdanov, A., Knežević, M., Mendel, F., Wang, Q.: FIDES: lightweight authenticated cipher with side-channel resistance for constrained hardware. In: Bertoni, G., Coron, J.-S. (eds.) CHES 2013. LNCS, vol. 8086, pp. 142–158. Springer, Heidelberg (2013)
6. Bogdanov, A., Rijmen, V.: Zero-correlation linear cryptanalysis of block ciphers. IACR Cryptology ePrint Archive 2011, 123 (2011). http://eprint.iacr.org/2011/123
7. Chabaud, F., Vaudenay, S.: Links between differential and linear cryptanalysis. In: De Santis, A. (ed.) EUROCRYPT 1994. LNCS, vol. 950, pp. 356–365. Springer, Heidelberg (1995)
8. Cid, C., Murphy, S., Robshaw, M.: Small scale variants of the AES. In: Gilbert, H., Handschuh, H. (eds.) FSE 2005. LNCS, vol. 3557, pp. 145–162. Springer, Heidelberg (2005)
9. Daemen, J., Govaerts, R., Vandewalle, J.: Correlation matrices. In: Preneel, B. (ed.) FSE 1994. LNCS, vol. 1008, pp. 275–285. Springer, Heidelberg (1995)
10. Daemen, J., Knudsen, L.R., Rijmen, V.: The block cipher SQUARE. In: Biham, E. (ed.) FSE 1997. LNCS, vol. 1267, pp. 149–165. Springer, Heidelberg (1997)
11. Daemen, J., Rijmen, V.: AES Proposal: Rijndael (1998)
12. Daemen, J., Rijmen, V.: The Design of Rijndael: AES - The Advanced Encryption Standard. Information Security and Cryptography. Springer, Heidelberg (2002). doi:10.1007/978-3-662-04722-4
13. Daemen, J., Rijmen, V.: Understanding two-round differentials in AES. In: De Prisco, R., Yung, M. (eds.) SCN 2006. LNCS, vol. 4116, pp. 78–94. Springer, Heidelberg (2006)
14. Daemen, J., Rijmen, V.: Plateau characteristics. IET Inf. Secur. **1**(1), 11–17 (2007)
15. Dinu, D., Corre, Y.L., Khovratovich, D., Perrin, L., Großschädl, J., Biryukov, A.: Triathlon of lightweight block ciphers for the internet of things. In: Lightweight Cryptography Workshop 2015 (2015)

16. Gao, Y., Guo, G.: Unified approach to construct 8 × 8 binary matrices with branch number 5. In: CDEE, pp. 413–416. IEEE (2010)
17. Gauravaram, P., Knudsen, L.R., Matusiewicz, K., Mendel, F., Rechberger, C., Schläffer, M., Thomsen, S.S.: Grøstl. a SHA-3 candidate (2011). http://groestl.info/specification.html
18. Gilbert, H., Peyrin, T.: Super-sbox cryptanalysis: improved attacks for AES-like permutations. In: Hong, S., Iwata, T. (eds.) FSE 2010. LNCS, vol. 6147, pp. 365–383. Springer, Heidelberg (2010)
19. Guo, J., Peyrin, T., Poschmann, A.: The PHOTON family of lightweight hash-functions. In: Rogaway, P. (ed.) CRYPTO 2011. LNCS, vol. 6841, pp. 222–239. Springer, Heidelberg (2011)
20. Guo, J., Peyrin, T., Poschmann, A., Robshaw, M.: The LED block cipher. In: Preneel, B., Takagi, T. (eds.) CHES 2011. LNCS, vol. 6917, pp. 326–341. Springer, Heidelberg (2011)
21. Kavun, E.B., Lauridsen, M.M., Leander, G., Rechberger, C., Schwabe, P., Yalçın, T.: Prøst. CAESAR Proposal (2014). http://proest.compute.dtu.dk
22. Knudsen, L.R.: Practically secure Feistel ciphers. In: Anderson, R. (ed.) FSE 1993. LNCS, vol. 809, pp. 211–221. Springer, Heidelberg (1994)
23. Knudsen, L.R., Wagner, D.: Integral cryptanalysis. In: Daemen, J., Rijmen, V. (eds.) FSE 2002. LNCS, vol. 2365, pp. 112–127. Springer, Heidelberg (2002)
24. Koo, B.-W., Jang, H.S., Song, J.H.: Constructing and cryptanalysis of a 16 × 16 binary matrix as a diffusion layer. In: Chae, K.-J., Yung, M. (eds.) WISA 2003. LNCS, vol. 2908, pp. 489–503. Springer, Heidelberg (2004)
25. Kwon, D., Sung, S.H., Song, J.H., Park, S.: Design of block ciphers and coding theory. Trends Math. **8**(1), 13–20 (2005)
26. Lai, X., Massey, J.L.: Markov ciphers and differential cryptanalysis. In: Davies, D.W. (ed.) EUROCRYPT 1991. LNCS, vol. 547, pp. 17–38. Springer, Heidelberg (1991)
27. Matsui, M.: On correlation between the order of S-boxes and the strength of DES. In: De Santis, A. (ed.) EUROCRYPT 1994. LNCS, vol. 950, pp. 366–375. Springer, Heidelberg (1995)
28. Mouha, N., Wang, Q., Gu, D., Preneel, B.: Differential and linear cryptanalysis using mixed-integer linear programming. In: Wu, C.-K., Yung, M., Lin, D. (eds.) Inscrypt 2011. LNCS, vol. 7537, pp. 57–76. Springer, Heidelberg (2012)
29. Nyberg, K.: Linear approximation of block ciphers. In: De Santis, A. (ed.) EUROCRYPT 1994. LNCS, vol. 950, pp. 439–444. Springer, Heidelberg (1995)
30. Rijmen, V., Daemen, J., Preneel, B., Bosselaers, A., De Win, E.: The cipher SHARK. In: Gollmann, D. (ed.) FSE 1996. LNCS, vol. 1039, pp. 99–111. Springer, Heidelberg (1996)
31. Sasaki, Y., Todo, Y., Aoki, K., Naito, Y., Sugawara, T., Murakami, Y., Matsui, M., Hirose, S.: Minalpher. CAESAR Proposal (2014). http://info.isl.ntt.co.jp/crypt/minalpher/index.html