Journal of Computer Science and Cybernetics, V.40, N.2 (2024), 187–203
DOI no. 10.15625/1813-9663/18058

ENHANCING THE SECURITY OF AES BLOCK CIPHER BASED ON MODIFIED MIXCOLUMN TRANSFORMATION

LUONG TRAN THI

Academy of Cryptography Techniques, No.141 Chien Thang, Tan Trieu, Thanh Tri, Ha Noi, Viet Nam

Abstract. Block ciphers, particularly Substitution-Permutation Network (SPN) block ciphers such as AES, are extensively utilized in contemporary cryptography. However, they face strong cryptanalysis including differential, linear, and algebraic cryptanalysis. Hence, enhancing the security of block ciphers, particularly AES, is a pressing research area. Besides security, the execution cost of block ciphers is crucial. This paper elucidates how Maximum Distance Separable (MDS) matrices enhance the diffusion layer's branch number in block ciphers, boosting their security. We propose a method to enhance AES security by altering its Mixcolumn transformation using efficient MDS matrices of various sizes. Additionally, we devise a technique to evaluate the fixed point coefficients of D(A) and fixed points in the modified AES diffusion layers. We demonstrate the branch number of modified AES diffusion layers with MDS matrices of sizes 8 and 16, analyzing their security, statistical standards, and execution speed. Our findings indicate a significant enhancement in AES security through our proposed approach.

Keywords. MDS matrix; Modified mixcolumn transformation; AES.

1. INTRODUCTION

Thanks to the results over the last 25 years on Substitution-Permutation Network (SPN) block ciphers, these block ciphers are also rated as "provable secure" against two basic attacks as differential attacks [1, 2] and linear attacks [3, 4]. After the proposal of Rijndael by Daemen and Rijmen was chosen as Advanced Ecnryption Standard (AES) [5, 6], the research on SPN block ciphers has been interested and developed very quickly. An SPN block cipher typically comprises three components: the substitution transformation typically employs substitution boxes (S-boxes), the diffusion transformation typically employs MDS matrices, and the key addition transformation.

From 1999-2002, Daemen and Rijmen [5, 6] coined the phrase Wide trail strategy to describe a comprehensive design approach, encompassing the selection of a linear mapping aimed at maximizing a significant quantity of active S-boxes (which are S-boxes having non-zero inputs). The two authors showed that the minimum number of active S-boxes of AES in any 4-round differential characteristics or 4-round linear characteristics is 25. After the AES authors' study, in his thesis in 2003 [7], Keliher also showed results regarding the practical

Corresponding author.
E-mail addresses: luongtranhong@gmail.com (L.T. Thi).

security of SPN block ciphers. However, the results of Keliher's theoretical evaluation are somewhat "raw" than those of the AES authors.

Vaudenay initially suggested the application of MDS matrices for linear conversions and later included them in the SHARK block cipher, followed by the SQUARE block cipher. This category of linear conversion offers the benefit of ensuring that the least number of activated S-boxes in two successive rounds of a differential characteristic is maximally equivalent to n+1 (where n is the number of S-boxes in one round of the SPN). That is why the diffusion layers of many current block ciphers use MDS matrices such as AES [5, 6], Shark, Khazad, Hierocrypt, Twofish, Square, Anubis, so on. MDS matrices are additionally incorporated into the construction of hash functions, exemplified by Maelstrom, Grøstl, and the lightweight hash function family Photon, which similarly employ MDS matrices as the primary element within their diffusion layers. There are many ways to construct the MDS matrix and the simplest way is to use MDS codes. Furthermore, numerous alternative approaches have been investigated for producing MDS matrices, including techniques derived from Hadamard matrices [8], Cauchy matrices [9], and Vandermonde matrices [10], recursive MDS matrices [11, 12], circulant and circulant-like matrices [13, 14], so on.

Presently, various dimensions of MDS matrices are employed in numerous established block ciphers, such as the circulant MDS matrices with a dimension of 4 over $GF(2^8)$ were used in the AES, a block cipher standard of NIST in 2000, MDS matrices of size 8 over $GF(2^8)$ are used in the Kalyna as a standard of Ukrainian [15] in 2015, involutory MDS matrices of size 8 over $GF(2^8)$ are used in Khazad block ciphers, 16×16 recursive MDS matrices over $GF(2^8)$ are used in the Russian 2015 GOST R34.12-2015 [16] standard. The study of AES block cipher modification has been interesting and numerous investigations have focused on rendering the substitution layer of AES adaptable [17, 18] or the diffusion layer [19, 20]. In addition to the direction of making the AES dynamic, the improvement of the AES Mixcolumn transformation direction is also of interest. In [21], the authors proposed to use a 4×4 involutory MDS Hadamard matrix to replace AES's 4×4 circulant MDS matrix and showed the efficiency in the software implementation of this matrix. In [22], the authors proposed to use a 8×8 involutory MDS Hadamard matrix found by an exhaustive method. The authors proposed to use this matrix as a replacement for AES's 4×4 circulant matrix, and provided a comparison of cycles and code memory compared to AES. In [23], the authors proposed a 16×16 involutory MDS matrix built from a Cauchy matrix. They reported that they constructed a 16×16 involutory MDS matrix to replace the diffusion layer matrix of AES. And the authors called the modified AES block cipher of MDS-AES. However, the authors in [24] proved that this is not an MDS matrix because some its submatrices are singular. In general, the above proposals to improve AES's Mixcolumn transformation only focus on involutory MDS matrix forms such as Hadamard matrix or Cauchy matrix without relating to other types of MDS matrix. On the other hand, some works have not given a clear assessment of the modified AES block cipher after including these MDS matrices. These evaluations encompass the security, statistical criteria, and operational velocity of the altered AES block cipher. In [25], we presented effective MDS matrices for performance with three MDS matrix types: Type-I circulant-like MDS matrices, Hadamard MDS matrices, and Recursive MDS matrices. The proposed matrices have sizes of 4, 8, and 16, and we also compare the number of Xtime and XOR operations of the proposed matrices with the MDS matrices of famous ciphers such as AES, Square, Twofish, Hierocrypt, Khazad, and GOST

R34.12-2015. The results in [25] are a crucial basis for us to use and apply them in this paper to modify the Mixcolumn transformation of the AES block cipher.

In this paper, we clarify the role of the MDS matrix in increasing the branch number of the diffusion layer of the block ciphers, thereby improving the security of the block ciphers. We present an approach to enhance the security of the AES block cipher by replacing the mixcolumn operation of AES with MDS matrices of sizes 4, 8, or 16, which are optimized for efficiency in implementation. We present a method to find a new diffusion matrix of modified AES block ciphers from which to evaluate the fixed points coefficient D(A) and number of fixed points of the modified AES diffusion layers. In addition, we prove the branch number of the modified AES diffusion layers with MDS matrices of sizes 8, and 16. Subsequently, we conduct an evaluation of the security, statistical metrics, and operational efficiency of the adapted AES block ciphers derived from these MDS matrices. The findings indicate a substantial enhancement in the security of the AES block cipher through our proposed methodology. The structure of this paper is outlined as follows: Section 2 introduces foundational concepts and relevant literature. Section 3 delineates the alteration of the Mixcolumn transformation in AES through the utilization of MDS matrices with dimensions 4, 8, and 16. Section 4 assesses the performance of the adapted AES block ciphers. Finally, Section 5 provides the concluding remarks.

2. PRELIMINARIES AND RELATED WORKS

2.1. MDS code and MDS matrix

Within the realm of error-correcting code theory, for every linear code $C[n, k, d]$, denoted by n as the length, k as the dimensions, and d as the minimum distance, there exists a constraint known as the singleton bound $d \leq n - k + 1$ [26, Theorem 11]. If the minimum distance d equals $n - k + 1$, the code is termed as an MDS code. This theory encompasses a significant theorem in coding theory.

Theorem 1. *([26, Theorem 8]) A code [n,k,d] with a generator matrix G=[I —A] and A being a matrix of size k×(n-k), is an MDS code if and only if every possible square submatrix of A is non-singular. The matrix A is referred to as the MDS matrix.*

2.2. Branch number

For an invertible linear transformation F, its diffusion measure is given as follows.

Definition 1. [27] The branch number of the linear mapping $F : (GF(2^m))^n \to (GF(2^m))^n$ is defined by $\beta = \min_{a \neq 0}(W(a) + W(F(a)))$, where $W(a)$ represents the count of non-zero elements (or locations) within the vector $a \in (GF(2^m))^n$.

Every element of a serves as the input to the dispersal layer (and functions as the output of the substitution layer), while $F(a)$ denotes the output of this dispersal layer (the subsequent surrogate's input), thereby rendering the branch number as the least count of activated S-boxes functioning across two successive rounds of an SPN block cipher.

Note that the active S-boxes are those with non-zero inputs. According to [27], β is a measure of the diffusivity in the sense that the larger the β, the better the diffusivity.

In [7], Keliher evaluated the actual security of the SPN block cipher against differential and linear cryptanalysis. Assume that $\Omega = (a^1, a^2, \ldots, a^T, a^{T+1})$ is a-round linear character-

istic. In the implementation of the linear attack, the assailant employs a direct exploration technique to identify a T-round linear characteristic $\widehat{\Omega}$ wherein $ELCP(\widehat{\Omega})$ is optimized. That characteristic needs not to be unique and it is called the best characteristic. Then, one considers the data complexity (number of plaintext/ciphertext pairs) according to Matsui's algorithm [3] against linear attacks as follows

$$N_L \approx \frac{c}{ELCP(\widehat{\Omega}))}, \tag{1}$$

where, the value of $ELCP(\widehat{\Omega}))$ is called the mean linear characteristic probability of the characteristic Ω. Thus, the smaller this value is, the greater the data complexity of the linear cryptanalysis on the SPN block cipher, that is, the more secure the SPN algorithm.

Comparable to differential attacks, the metric $EDCP(\Omega)$ is termed as the average differential characteristic probability of characteristic Ω. A decreased value of this metric indicates superior performance.

2.3. The number of fixed points and the coefficient of fixed points of a linear transformation

For a linear transformation $F : (GF(2^m))^n \to (GF(2^m))^n$. The number of fixed points of F is calculated according to the following formula

$$F_A = 2^{r(m-rank[A-I])}. \tag{2}$$

The fixed points coefficient of F is calculated according to the following formula

$$D(A) = \frac{1}{m2^{mr}} \sum_{l=0}^{m-1} F_{A_{(l)}} = \frac{1}{m2^{mr}} \sum_{l=0}^{m-1} 2^{r(m-rank[A-I_{(l)}])}. \tag{3}$$

See more in [28].

2.4. Efficient MDS matrices for performance in [25]

In [25], we proposed efficient MDS matrices for performance with matrix types including Type-I circulant MDS matrices, Hadamard MDS matrices, and Recursive MDS matrices of sizes 4, 8, and 16.

We evaluated the proposed matrices based on the fixed points coefficient, number of fixed points, number of Xtimes, and number of XORs. Suggested matrices include:

- Three matrices of size 4 include Type-I circulant MDS matrices, Hadamard MDS matrices, and Recursive MDS matrices
- Three matrices of size 8 include Type-I circulant MDS matrices, Hadamard MDS matrices, and Recursive MDS matrices.
- One matrix of size 16 is a recursive MDS matrix.

Details of these matrices can be found in [25]. We will use these matrices in this paper to modify AES's mixcolumn operation.

3. MODIFICATION OF AES'S MIXCOLUMN TRANSFORMATION WITH MDS MATRICES OF SIZE 4, 8, 16

3.1. The role of the MDS matrix with the branch number of the diffusion layer

Within Subsection 2.2, the parameter β denoting the branch number of a linear transformation F is introduced. β serves as a gauge of the dispersal capacity of F, where in higher values of β correspond to enhanced dispersal capabilities. Conversely, within the diffusion layer, when employing a linear transformation with an MDS matrix, the branch number of said linear transformation coincides with the minimum distance d of the MDS code associated with the respective MDS matrix. In the ensuing discussion, we will meticulously demonstrate and substantiate this assertion.

Proposition 1. *Assume C is an MDS code $[2n, n, n+1]$ over $GF(2^m)$ and $G = [I_{nn}|B_{nn}]$ is the echelon-form generator matrix of C, where B is a non-singular matrix of size n and I is an Identity matrix of size n. Then, C defines an optimal invertible linear map γ (that is, with the branch number $\beta = n + 1$).*

$$\gamma : GF(2^m)^n \to GF(2^m)^n : X \to Y = B^T \times X^T. \tag{4}$$

Proof. First, it is necessary to show the branch number of the linear transformation γ satisfying: $\beta = d$, where $d = n + 1$ is the minimum distance of the code C

$$B = \begin{pmatrix} b_{0,0} & b_{0,1} & ... & b_{0,n-1} \\ b_{1,0} & b_{1,1} & ... & b_{1,n-1} \\ ... & ... & ... & ... \\ b_{n-1,0} & b_{n-1,1} & ... & b_{n-1,n-1} \end{pmatrix}. \tag{5}$$

Then, matrix B^T will have the following form

$$B^T = \begin{pmatrix} b_{0,0} & b_{1,0} & ... & b_{n-1,0} \\ b_{0,1} & b_{1,1} & ... & b_{n-1,1} \\ ... & ... & ... & ... \\ b_{0,n-1} & b_{1,n-1} & ... & b_{n-1,n-1} \end{pmatrix}. \tag{6}$$

Suppose $x = (x_0, x_1, \ldots, x_{n-1})$ where $x_i \in GF(2^m)$, $0 \le i \le n-1$, is the input of the linear transformation ω represented by B^T. Then the output of this linear transformation is

$$y = B^T x^T = \begin{bmatrix} b_{0,0} & b_{1,0} & ... & b_{n-1,0} \\ b_{0,1} & b_{1,1} & ... & b_{n-1,1} \\ ... & ... & ... & ... \\ b_{0,n-1} & b_{1,n-1} & ... & b_{n-1,n-1} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ ... \\ x_{n-1} \end{bmatrix} = \begin{bmatrix} \sum_{i=0}^{n-1} b_{i,0} x_i \\ \sum_{i=0}^{n-1} b_{i,1} x_i \\ ... \\ \sum_{i=0}^{n-1} b_{i,n-1} x_i \end{bmatrix} = \begin{bmatrix} y_0 \\ y_1 \\ ... \\ y_{n-1} \end{bmatrix}. \tag{7}$$

According to Definition 1, the branch number of this linear transform is

$$\beta = min_{x \neq 0}(W(x) + W(B^T x^T)) = min_{x \neq 0}(W(x) + W(y)) \tag{8}$$

where y is calculated by the formula (7).

Going back to the MDS code $C[2n, n, n+1]$ and the echelon-form generator matrix $G = [I_{nn}|B_{nn}]$. Given an input message includes n components of the form $x = (x_0, x_1, \ldots, x_{n-1})$, the resulting codeword of this code (with a length of $2n$ components) is:

$$\begin{aligned}\overline{y} = xG = (x_0, x_1, ..., x_{n-1}) &\begin{bmatrix} 1 & ... & ... & ... & 0 & b_{0,0} & b_{0,1} & ... & b_{0,n-1} \\ 0 & 1 & ... & ... & 0 & b_{1,0} & b_{1,1} & ... & b_{1,n-1} \\ ... & ... & ... & ... & ... & ... & ... & ... & ... \\ 0 & 0 & ... & ... & 1 & b_{n-1,0} & b_{n-1,1} & ... & b_{n-1,n-1} \end{bmatrix} \\ = \left(x_0, x_1, ..., x_{n-1}, \textstyle\sum_{i=0}^{n-1} b_{i,0}x_i, \sum_{i=0}^{n-1} b_{i,1}x_i, ..., \sum_{i=0}^{n-1} b_{i,n-1}x_i\right) \\ = (x_0, x_1, ..., x_{n-1}, y_0, y_1, ..., y_{n-1})\end{aligned} \tag{9}$$

Comparing the results of (7) and (9), see that $W(\overline{y}) = W(x) + W(y)$ where y is calculated by formula (7). Thus, the minimum distance of the code C is calculated by the formula

$$d = \min_{\overline{y\neq 0}, \overline{y}} \text{ is the codeword of } C^{W(\overline{y})} = \min_{x\neq 0}(W(x) + W(y)). \tag{10}$$

Comparing formulas (8) and (10), deduce that $d = \beta$. Also, with the above MDS code $C[2n, n, n+1]$, there is always $d = n + 1$. Therefore, $d = \beta = n + 1$.

Since B^T is non-singular, the map γ is invertible. ∎

Remark 1. *The proof in the above proposition shows that the branch number of this linear transformation is equal to the minimum distance of the MDS code $C[2n, n, n+1]$, i.e: $\beta = d$. From the singleton bound, it can be seen that the MDS code has the minimum distance reaching the maximum value. Therefore, the branch number of the above linear transformation also reaches the maximum possible value.*

Through Proposition 1, we can see the crucial role of the MDS matrix in creating the maximum branch number of the linear transformation using it. If the MDS matrix is of size n, then the linear transformation corresponding to this matrix will have the largest possible branch number and equal $n+1$. When this branch number is maximum, it means the linear transformation uses the MDS matrix with the largest possible diffusion. When this branch number is maximized, the block cipher's resistance to linear and differential attacks is also as strong as possible [7]. That is why MDS matrices have been chosen for the diffusion layer of many of today's well-known block ciphers and hash functions.

3.2. A method to find new diffusion matrices of modified AES block ciphers

In the case of modified AES block ciphers, MDS matrices with different sizes as 4×4, 8×8, 16×16 can be inserted to replace the AES 4×4 circulant MDS matrix. Therefore, the diffusion layer structure of modified AES and the way its multiplications perform are different from the original AES. In the following, we will present the new diffusion layer structure, and how to perform the multiplications, and from that, we can derive a new diffusion matrix of modified AES block ciphers to calculate the number of fixed points and the coefficient of fixed points $D(A)$ for those modified AES block ciphers.

The two diffusion layer transformations of the original AES remain unchanged, namely the linear L_0 transformation (ShiftRow transformation) and the linear L_1 transformation (MixColumn transformation). However, the MDS matrix in L_1 is replaced by another MDS matrix of size 4, 8, or 16.

Presume that the vector $x = (x_0, x_1, \ldots, x_{15})$ serves as the input to the modified AES diffusion layer where x_i is in $GF(2^8)$. Then, the output of this layer is denoted by $y = (y_0, y_1, \ldots, y_{15})$, where y_i is in $GF(2^8)$. The output of the L_0 transformation with input $x = (x_0, x_1, \ldots, x_{15})$ is $\widehat{y} = (\widehat{y}_0, \widehat{y}_1, ..., \widehat{y}_{15})$. Thus, it is to have

$$y = L_1(L_0(x)) = L_1(\widehat{y}). \tag{11}$$

Method of finding new 16×16 diffusion matrices of modified AES block ciphers.

In the following, we will present the steps to find a new 16×16 diffusion matrix of the modified AES and consider the different cases where the MDS matrices are used of size 4, 8, or 16 instead of the original Mixcolumn transformation in AES.

Step 1: The input x will be converted to a 4×4 state array as follows

$$\begin{bmatrix} x_0 & x_4 & x_8 & x_{12} \\ x_1 & x_5 & x_9 & x_{13} \\ x_2 & x_6 & x_{10} & x_{14} \\ x_3 & x_7 & x_{11} & x_{15} \end{bmatrix}. \tag{12}$$

Step 2: The state array x is passed through the transformation L_0 to $\widehat{y}$ has the following form

$$\begin{bmatrix} \widehat{y}_0 & \widehat{y}_4 & \widehat{y}_8 & \widehat{y}_{12} \\ \widehat{y}_1 & \widehat{y}_5 & \widehat{y}_9 & \widehat{y}_{13} \\ \widehat{y}_2 & \widehat{y}_6 & \widehat{y}_{10} & \widehat{y}_{14} \\ \widehat{y}_3 & \widehat{y}_7 & \widehat{y}_{11} & \widehat{y}_{15} \end{bmatrix}. \tag{13}$$

Where the application of the linear transform L_0 to x is described by the following formula (similar to that of the original AES)

$$\widehat{y} = \begin{cases} x_i & \text{for } i = 0, 4, 8, 12 \\ x_{i+4} & \text{for } i = 1, 5, 9, 13 \\ x_{i+8} & \text{for } i = 2, 6, 10, 14 \\ x_{i+12} & \text{for } i = 3, 7, 11, 15. \end{cases} \tag{14}$$

Step 3: $\widehat{y}$ is passed through the transformation L_1 to y with the following form

$$\begin{bmatrix} y_0 & y_4 & y_8 & y_{12} \\ y_1 & y_5 & y_9 & y_{13} \\ y_2 & y_6 & y_{10} & y_{14} \\ y_3 & y_7 & y_{11} & y_{15} \end{bmatrix}. \tag{15}$$

Depending on the size of the new MDS matrix (M) introduced into L_1, the matrix structure of $\widehat{y}$ will be changed to perform the multiplication between M and the matrix of $\widehat{y}$.

If the matrix M has a size of 4×4

$$\begin{bmatrix} m_{0,0} & m_{0,1} & m_{0,2} & m_{0,3} \\ m_{1,0} & m_{1,1} & m_{1,2} & m_{1,3} \\ m_{2,0} & m_{2,1} & m_{2,2} & m_{2,3} \\ m_{3,0} & m_{3,1} & m_{3,2} & m_{3,3} \end{bmatrix}, \tag{16}$$

then the transformation L_1 is performed as follows $y = M\widehat{y}$

$$M\widehat{y} = \begin{bmatrix} m_{0,0} & m_{0,1} & m_{0,2} & m_{0,3} \\ m_{1,0} & m_{1,1} & m_{1,2} & m_{1,3} \\ m_{2,0} & m_{2,1} & m_{2,2} & m_{2,3} \\ m_{3,0} & m_{3,1} & m_{3,2} & m_{3,3} \end{bmatrix} \times \begin{bmatrix} y_0 & y_4 & y_8 & y_{12} \\ y_1 & y_5 & y_9 & y_{13} \\ y_2 & y_6 & y_{10} & y_{14} \\ y_3 & y_7 & y_{11} & y_{15} \end{bmatrix} = \begin{bmatrix} y_0 & y_4 & y_8 & y_{12} \\ y_1 & y_5 & y_9 & y_{13} \\ y_2 & y_6 & y_{10} & y_{14} \\ y_3 & y_7 & y_{11} & y_{15} \end{bmatrix}. \quad (17)$$

Thus, through (17) it is possible to represent each y_i through y_i, combining this formula with the representation in (14), we can find a new diffusion matrix of size 16 of the modified AES in this case (the new MDS matrix is 4×4 instead of the MDS matrix in the AES diffusion layer). If the matrix M has a size of 8×8, and suppose $M = [m_{i,j}],\ i \neq 0,\ j \neq 7$.

Convert the matrix $\widehat{y}$ to the form of 8×2 instead of 4×4 so that it can be multiplied by the matrix M. Now, the matrix $\widehat{y}$ has the following form

$$\widehat{y} = \begin{bmatrix} \widehat{y}_0 & \widehat{y}_1 & \widehat{y}_2 & \widehat{y}_3 & \widehat{y}_4 & \widehat{y}_5 & \widehat{y}_6 & \widehat{y}_7 \\ \widehat{y}_8 & \widehat{y}_9 & \widehat{y}_{10} & \widehat{y}_{11} & \widehat{y}_{12} & \widehat{y}_{13} & \widehat{y}_{14} & \widehat{y}_{15} \end{bmatrix}^T. \quad (18)$$

Then, the transformation L_1 is performed as follows

$$M \times y = \begin{bmatrix} \widehat{y}_0 & \widehat{y}_1 & \widehat{y}_2 & \widehat{y}_3 & \widehat{y}_4 & \widehat{y}_5 & \widehat{y}_6 & \widehat{y}_7 \\ \widehat{y}_8 & \widehat{y}_9 & \widehat{y}_{10} & \widehat{y}_{11} & \widehat{y}_{12} & \widehat{y}_{13} & \widehat{y}_{14} & \widehat{y}_{15} \end{bmatrix}^T. \quad (19)$$

The resulting matrix at (19) will then be converted to a 4×4 form to get the matrix y at (15), which is the output of the linear transformation L_1. Thus from (19), it can represent each y_i through $\widehat{y}_j$ $(0 \leq j \leq 15)$, and combined with the representation in (14), we can find a new 16×16 diffusion matrix of the modified AES in this case.

If the matrix M has a size of 16×16, and suppose $M = [m_{i,j}],\ 0 \leq i, j \leq 15$.

Then, the matrix $\widehat{y}$ is converted to 16×1 instead of 4×4 so that it can be multiplied by the matrix M. Now, the matrix $\widehat{y}$ has the following form

$$\widehat{y} = [\widehat{y}_0\ \widehat{y}_1\ \widehat{y}_2\ \widehat{y}_3\ \widehat{y}_4\ \widehat{y}_5\ \widehat{y}_6\ \widehat{y}_7\ \widehat{y}_8\ \widehat{y}_9\ \widehat{y}_{10}\ \widehat{y}_{11}\ \widehat{y}_{12}\ \widehat{y}_{13}\ \widehat{y}_{14}\ \widehat{y}_{15}]^T. \quad (20)$$

Then, the transformation L_1 is performed as follows

$$\begin{aligned} M \times \widehat{y} &= M \times [\widehat{y}_0\ \widehat{y}_1\ \widehat{y}_2\ \widehat{y}_3\ \widehat{y}_4\ \widehat{y}_5\ \widehat{y}_6\ \widehat{y}_7\ \widehat{y}_8\ \widehat{y}_9\ \widehat{y}_{10}\ \widehat{y}_{11}\ \widehat{y}_{12}\ \widehat{y}_{13}\ \widehat{y}_{14}\ \widehat{y}_{15}]^T \\ &= [\widehat{y}_0\ \widehat{y}_1\ \widehat{y}_2\ \widehat{y}_3\ \widehat{y}_4\ \widehat{y}_5\ \widehat{y}_6\ \widehat{y}_7\ \widehat{y}_8\ \widehat{y}_9\ \widehat{y}_{10}\ \widehat{y}_{11}\ \widehat{y}_{12}\ \widehat{y}_{13}\ \widehat{y}_{14}\ \widehat{y}_{15}]^T. \end{aligned} \quad (21)$$

The resulting matrix at (21) will then be converted to 4×4 form to get the matrix γ at (15), which is the output of the linear transform L_1.

Thus, from (21) it can represent each y_i through $\widehat{y}_j$ $(0 \leq j \leq 15)$, combined with the representation in (14), we can find a new 16×16 diffusion matrix of the modified AES in this case.

Assessment of the quantity of fixed points and the coefficient $D(A)$ within the altered AES diffusion layer.

Upon discovering a novel 16×16 diffusion matrix for the adjusted AES (by the fresh MDS matrix implemented within the MixColumn transformation of AES), we can compute the fixed-point count and the $D(A)$ coefficient of the adapted AES diffusion layer. The calculation of fixed-point count and the $D(A)$ coefficient adheres to the equations outlined in Subsection 2.3.

In reference [25], a detailed assessment was conducted on the quantity of stationary points and the $D(A)$ coefficient of the adjusted AES diffusion layer following the application of the suggested MDS matrices with dimensions of 4, 8, and 16.

3.3. Branch number of the modified AES diffusion layer

Using the MDS matrices suggested in [25], we sequentially integrate these matrices into the diffusion layer of AES, substituting the 4×4 circulant MDS matrix of AES. Consequently, the architecture of the diffusion layer in the modified AES block cipher undergoes alterations based on the dimensions of the input MDS matrix. As shown in Subsection 3.2, in the diffusion layer of the modified AES block cipher, the ShiftRow transformation still behaves the same as in the original AES, only the multiplication in the MixColumn transformation changes depending on the size of the newly introduced MDS matrix. Also note that any other components of AES as substitution layer, key scheme, key size, number of encryption rounds, and so on, remain the same as the original AES.

For the new 4×4 MDS matrices, the multiplication performed on the MixColumn transformation is still performed regularly like the multiplication in the original AES MixColumn. At this point, denote the new AES block cipher is AES4. For the new MDS matrices of size 8×8, 16×16, the multiplication in the MixColumn transformation changes. In these cases, the new AES block ciphers are denoted by AES8 and AES16, respectively.

Next, we give the following proposition to confirm that the new diffusion layer of AES8 and AES16 still ensures linearity and has a branch number of 9 or 17.

Proposition 2. *When introducing an 8×8 matrix or a 16×16 matrix into the mixcolumn transformation of the AES block cipher, the composite transformation of the diffusion layer of the modified AES is a linear transformation with the branch numbers respectively 9 or 17.*

Proof. Consider the case of a new MDS matrix of size 8×8. Let's consider substituting the 4×4 matrix within the Mixcolumn transformation of AES with an MDS matrix -sized 8×8, denoted as $M = [m_{i,j}]_{88}$. Let's assume that the state array x (with a size of 4×4) serves as the input to the AES8 diffusion layer.

The AES diffusion layer consists of two transformations, ShiftRow and Mixcolumn. However, the diffusion layer of AES8 will include the following four transformations:

- Transformation 1 (denoted L_1) is the ShiftRow transformation of AES.
- Transformation 2 (denoted L_2): Converts the input state array x of size 4×4 to an array of size 8×2 (see (18)).
- Transformation 3 (denoted L_3): Multiply the 8×2 array above by the matrix M (similar to the operation of the AES mixcolumn transformation) to get an 8×2 array.
- Transformation 4 (denoted by L_4): Converts the resulting 8×2 array of L_3 to a 4×4 array (see (19)).

Thus, the overall transformation (denoted L) of the diffusion layer of AES8 is a combination of the above four transformations. Assume that for an input state array x of size 4×4, its output through L is denoted by x'. Then, get

$$x' = L(x) = L_4(L_3(L_2(L_1(x)))). \tag{22}$$

Since L_1 is the original AES ShiftRow transformation, L_1 is a linear transformation. The L_3 transformation is a matrix multiplication, so L_3 is also a linear transformation. With the

two transformations L_2 and L_4, it is easy to see that they are both linear transformations. Therefore L is a linear transform.

Now, we will find the branch number of L, denoted $\beta(L)$. Again, the branch number $\beta(L)$ of a linear transformation L is given by the following formula

$$\beta(L) = \min_{x \neq 0}\{W(x) + W(L(x))\}. \tag{23}$$

To find the branch number of L, according to the definition of the branch above, we will consider in turn 8 cases of the input array x (size of 4×4) of the linear transformation L (x has 1 non-zero byte, 2 non-zero bytes, 8 non-zero bytes). In each such case, find the number of non-zero bytes of the output $L(x)$. For simplicity and the proof in general, suppose that x is any input array of L and that x contains u non-zero bytes ($1 \leq u \leq 8$). Assuming y is the output of x through L, then it is to have

$$y = L(x) = L_4(L_3(L_2(L_1(x)))). \tag{24}$$

Suppose $\widehat{y}_1 = L_2(L_1(x))$, $\widehat{y}_2 = L_3(L_2(L_1(x))) = L_3(\widehat{y}_1)$. It is to have

$$y = L_4(\widehat{y}_2). \tag{25}$$

Since the linear transformations L_1, L_2 do not change the number of non-zero bytes of x, the number of non-zero bytes of $\widehat{y}_1$ is still equal to u. Or

$$W(x) = W(\widehat{y}_1) = u. \tag{26}$$

According to Proposition 1, the branch number of the linear transformation L_3 (represented by the MDS matrix M of size 8×8) will be 9. Thus, it is to have

$$W(\widehat{y}_1) + W(\widehat{y}_2) \geq 9. \tag{27}$$

Since the L_4 transformation does not change the number of non-zero bytes of $\widehat{y}_2$, we get

$$W(\widehat{y}_2) = W(y). \tag{28}$$

By (26), (27), and (28) it deduces

$$W(x) + W(y) \geq 9. \tag{29}$$

By (23) and (29), it is to have $\beta(L) = 9$. Thus, the branch number of the diffusion layer of AES8 is 9. Consider the case of a new MDS matrix of size 16×16 For this case, the proof is similar to the case of a new MDS matrix of size 8×8 with:

- Transformation 1 (denoted L_1) is the ShiftRow transformation of AES.
- Transformation 2 (denoted L_2): Converts the input state array x of size 4×4 to an array of size 16×1 (see (20)).
- Transformation 3 (denoted L_3): Multiply the 16×1 array above by the matrix M (similar to the operation of the AES mixcolumn transformation) to get a 16×1 array.
- Transformation 4 (denoted by L_4): Converts the resulting 16×1 array of L_3 to a 4×4 array (see (21)).

As a result, the branch number of the diffusion layer of AES16 is 17. ■

In Subsection 4.3, we will compare the performance of modified AES block ciphers when introducing new MDS matrices into the AES diffusion layer of different types and sizes.

4. EVALUATION OF MODIFIED AES BLOCK CIPHERS

4.1. Security analysis

In this section, the modified AES block ciphers will be evaluated for actual security against linear and differential attacks according to Keliher's evaluation formulas [7]. As detailed in Subsection 3.2, various MDS matrices of dimensions 4×4, 8×8, and 16×16, as suggested in [25], have been incorporated into the Mixcolumn transformation instead of the 4×4 MDS matrix utilized in AES. The branch numbers associated with these matrices are 5, 9, and 17, respectively, aligning with the branch number of the modified AES block ciphers' diffusion layer (as demonstrated in Proposition 2). Drawing from Keliher's findings, we will assess the upper limit of the maximum mean linear characteristic probability (ELCP) and the maximum mean differential characteristic probability (EDCP) to ascertain the data complexity associated with these attacks. Tables 1, 2, and 3 showcase the minimum count of operational S-boxes and the peak ELCP and EDCP values of the adapted AES block ciphers (incorporating the fresh MDS matrices of different dimensions, following Keliher's evaluation equations). Analogous to the AES block cipher, the adjusted AES block ciphers have iteration counts of 10, 12, and 14, correspondingly, aligning with key lengths of 128, 192, and 256 bits.

Table 1: Number of active S-boxes and upper bound of maximum ELCP and maximum EDCP of AES4 and AES with 4×4 MDS matrices ($\beta = 5$).

Round	The lower bound of the number of linear active S-boxes	The lower bound of the number of differential active S-boxes	Upper bound of maximum ELCP	Upper bound of maximum EDCP
10	21	21	2^{-126}	2^{-126}
12	26	26	2^{-156}	2^{-156}
14	31	31	2^{-186}	2^{-186}

Table 2: Number of active S-boxes and upper bound of maximum ELCP and maximum EDCP of AES4 and AES with 8×8 MDS matrices ($\beta = 9$).

Round	The lower bound of the number of linear active S-boxes	The lower bound of the number of differential active S-boxes	Upper bound of maximum ELCP	Upper bound of maximum EDCP
10	37	37	2^{-222}	2^{-222}
12	46	46	2^{-276}	2^{-276}
14	55	55	2^{-330}	2^{-330}

In the context of linear cryptanalysis, the adversary employs a straightforward exploration algorithm to discover a characteristic T rounds Ω where $ELCP(\Omega)$ attains its peak. In such scenarios, the data complexity is regarded as (1) where c being a small constant. For differential cryptanalysis, the data complexity is

$$N_D \approx \frac{c}{EDCP(\Omega)}. \tag{30}$$

Table 3: Number of active S-boxes and upper bound of maximum ELCP and maximum EDCP of AES4 and AES with 16×16 MDS matrices ($\beta = 17$).

Round	The lower bound of the number of linear active S-boxes	The lower bound of the number of differential active S-boxes	Upper bound of maximum ELCP	Upper bound of maximum EDCP
10	69	69	2^{-414}	2^{-414}
12	86	86	2^{-516}	2^{-516}
14	103	103	2^{-618}	2^{-618}

Moreover, for an SPN block cipher consisting of T rounds, it is to have [7]

$$EDCP(\Omega) \leq \begin{cases} q^{\beta_l (T/2)} & \text{if } T \text{ is even,} \\ q^{\beta_l \lfloor T/2 \rfloor + 1} & \text{if } T \text{ is odd.} \end{cases} \tag{31}$$

The formulas are similar for differential cryptanalysis as well.

Evaluating the S-boxes is crucial for determining the upper bound of *ELCP*, *EDCP* from (31), and simultaneously from (1) and (30) determining the complexity of linear and differential cryptanalysis.

The upper bounds of the values *ELCP* and *EDCP* shown in Tables 1, 2, and 3 also correspond to the respective data complexities of linear cryptanalysis and differential cryptanalysis.

Remark 2. *The actual security of the modified AES block ciphers (according to Keliher's assessment) is very high. Compared with AES, it can be seen that the actual security of AES4 is equal to that of AES, but the actual security of AES8 and AES16 is much higher than that of AES. For example, when the number of rounds is 10, the security of AES4 is* 2^{-126}*, but that of AES8 is* 2^{-222} *and AES16 is* 2^{-414}*.*

Thus, if we use efficient MDS matrices of size 8 and 16 for implementation, the branch number of AES's diffusion layer is increased from 5 to 9 and 17 respectively, which means the security of the modified AES block ciphers is greatly increased. In terms of speed, of course, modified AES block ciphers can be slightly slower than AES. The speed assessment will be presented in Subsection 4.3.

4.2. Evaluation of statistical standards of modified AES block ciphers

Table 4: NIST statistical criteria test for modified AES block ciphers (Sign ✓: indicates pass).

Cipher \ Test (CR)	1	2	3	4	5	6	7	8	9	10	11	12	13
AES	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
AES4, AES8, AES16	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

In the evaluation of NIST statistical standards, we use a random source obtained from the website "www.random.org". With the modified AES block ciphers, test for randomness according to several NIST statistical standards, including Frequency mono bit test (CR1),

Frequency test within a Block (CR2), Cumulative sums test (CR3), Runs test (CR4), Test for the longest run of ones in a block (CR5), Binary matrix rank test (CR6), Non-overlapping template matching test (CR7), Overlapping template matching test (CR8), Approximate entropy test (CR9), Random visiting test (CR10), Random excursions test (CR11), Serial test (CR12), and Linear complexity test (CR 13).

Table 5: Performance evaluation of AES4 block ciphers with 4×4 MDS matrices

		G1		G2			
C1	C2	C3	C4	C5	C6	C7	C8
Hadamard matrix	//0x169 0x47 0x20 0x60 0x6 0x20 0x47 0x6 0x60 0x60 0x6 0x47 0x20 0x6 0x60 0x20 0x47	128	886.125	128	59.590	64	48
		192	764.159	192	59.312		
		256	666.213	256	58.203		
Type-I circulant-like matrix	//0x169 0x02 0x01 0x01 0x01 0x01 0x01 0x28 0x02 0x01 0x02 0x01 0x28 0x01 0x28 0x02 0x01	128	886.09	128	59.75	64	48
		192	748.410	192	59.780		
		256	661.732	256	58.776		
Recursive matrix	//0x169 0x42 0xe2 0x01 0x63 0x7a 0x77 0x81 0x46 0xea 0x6f 0x31 0x1e 0xca 0xde 0x71 0x9e	128	886.09	128	59.795	64	48
		192	764.119	192	59.437		
		256	666.190	256	58.536		
Orgirinal AES matrix	//0x11B 0x02 0x03 0x01 0x01 0x01 0x02 0x03 0x01 0x01 0x01 0x02 0x03 0x03 0x01 0x01 0x02	128	886.510	128	59.999	64	48
		192	753.310	192	59.821		
		256	669.234	256	58.813		

The results obtained from the analysis in Table 4 indicate that both the AES block cipher and the altered AES block ciphers adhere to the aforementioned 13 standards. It's worth noting that the alternative MDS matrices of dimensions 4×4, 8×8, and 16×16 outlined in [25] have been integrated into the Mixcolumn operation instead of the 4×4 MDS matrix used in AES.

4.3. Evaluation of the speed of modified AES block ciphers

This section provides an assessment of the performance of modified AES block ciphers when introducing new MDS matrices of different types and sizes. Computer configuration used for testing includes Intel Core i3-4210M Processor CPU 2.6GHz, Internal Memory 4GB RAM, Microsoft Windows 7 Profesional 32-bit SP1. To compare the speed for 4×4 MDS matrices, in addition to the standard settings as in [5] by Vincent Rijmen (August 2001), we approach the efficient setting according to the optimized method. It is proposed by the authors of AES by combining the transformations of the round function (SubByte, ShiftRow, MixColumn) into lookup tables [6]. This method is implemented by Brian Gladman (the author responsible for implementing NIST's AES algorithm). In general, with this imple-

Table 6: Performance evaluation of AES8 block ciphers with 8×8 MDS matrices

C1	C2	C3	C4	C5	C6
Hadamard matrix	//0x169 0x64 0xc4 0x02 0x13 0x39 0x88 0x0b 0x0a 0xc4 0x64 0x13 0x02 0x88 0x39 0x0a 0x0b 0x02 0x13 0x64 0xc4 0x0b 0x0a 0x39 0x88 0x13 0x02 0xc4 0x64 0x0a 0x0b 0x88 0x39 0x39 0x88 0x0b 0x0a 0x64 0xc4 0x02 0x13 0x88 0x39 0x0a 0x0b 0xc4 0x64 0x13 0x02	128	55.457	128	112
	0x0b 0x0a 0x39 0x88 0x02 0x13 0x64 0xc4	192	55.164		
	0x0a 0x0b 0x88 0x39 0x13 0x02 0xc4 0x64	256	53.913		
Type-I circulant-like matrix	//0x169 0x04 0x01 0x01 0x01 0x01 0x01 0x01 0x01 0x01 0x01 0x05 0x14 0x85 0x84 0x7f 0x04 0x01 0x04 0x01 0x05 0x14 0x85 0x84 0x7f 0x01 0x7f 0x04 0x01 0x05 0x14 0x85 0x84 0x01 0x84 0x7f 0x04 0x01 0x05 0x14 0x85 0x01 0x85 0x84 0x7f 0x04 0x01 0x05 0x14	128	55.860	86	112
	0x01 0x14 0x85 0x84 0x7f 0x04 0x01 0x05	192	54.856		
	0x01 0x05 0x14 0x85 0x84 0x7f 0x04 0x01	256	54.116		
Recursive matrix	//0x169 0x61 0x64 0x70 0xc4 0xd4 0x54 0x82 0x47 0x72 0x21 0xec 0x6f 0x66 0x3a 0x96 0x18 0xbc 0xb6 0x6c 0xad 0xc7 0xf0 0x34 0xc1 0x25 0xe7 0x7c 0x72 0x9d 0xee 0x02 0x19 0xdd 0x85 0xda 0xf9 0x0e 0x5f 0x62 0x12 0xcd 0x4a 0x13 0x45 0x2f 0x42 0xe1 0x51	128	55.534	128	112
	0x52 0xe3 0xfd 0x38 0xda 0xb4 0xe4 0x3d	192	54.511		
	0x2e 0xb5 0x9b 0xef 0x81 0x80 0x2d 0x92	256	53.996		

mentation, since the transformations of the round function have been converted to lookup table form, the encryption/decryption speed when using different matrices of the same size 4×4 is equivalent and equivalent to using the original AES MDS matrix. The performance evaluation of AES4 block ciphers is shown in Table 5, C1: Matrix type, C2: Matrix representation 4×4, C3: Key length (bits), C4: Encyption /Decryption speed (Mb/sec), C5: Key length (bits), C6: Encyption/ Decryption speed Of AES4s (Mb/sec), C7: Number of multiplications (look up table)/1 round, C8: Number of additions/1 round, G1: Optimization implementation by Brian Gladman, G2: Standard implementation.

For the 8×8 and 16×16 MDS matrices, conduct experiments involving the incorporation of these matrices based on the functions corresponding to the algorithm's operations – the multiplication operations with the MDS matrix elements are carried out by referencing the finite field multiplication table. This installation is approached according to the standard installation of Vincent Rijmen [5]. The compilation environment is Microsoft Visual Studio 2015. The test results with all three key lengths 128, 192 and 256 bits and with different new MDS matrix sizes are described in Tables 5, 6, and 7. The performance evaluation of AES16 block ciphers is shown in Table 6, C1: Matrix type, C2: Matrix representation 8x8, C3: Key length (bits), C4: Encyption/ Decryption speed Of AES8s (Mb/sec), C5: Number of multiplications (look up table)/1 round, C6: Number of additions/1 round. In Table 7,

C1: Matrix type, C2:Matrix representation 16x16, C3: Key length (bits), C4: Encyption/ Decryption speed Of AES16s (Mb/sec), C5: Number of multiplications (look up table)/1 round, C6: Number of additions/1 round.

Remark 3. *From Tables 5, 6, and 7, it can be seen that the performance speed of AES4 is equivalent to the original AES. The performance speed of AES8, and AES16 is not significantly slower than the original AES when using the installation method according to the lookup tables. However, in return, modified block ciphers AES8, and AES16 provide much higher security than AES, especially for two strong attacks on block ciphers, linear attack, and differential attack.*

Table 7: Performance evaluation of AES16 block ciphers with 16×16 MDS matrices

C1	C2	C3	C4	C5	C6
Hadamard matrix	//0x169 4e a9 54 75 c7 61 62 d0 45 3d d6 1c 26 16 90 4e db f9 61 88 43 67 13 fa 7b 8f 68 08 8b c6 3a 4b c4 b3 5c 75 de 6f 96 a0 69 78 ef da 21 25 e8 fe ad e5 68 b3 33 17 cd 9d e0 fd a5 4f c6 92 d3 45 ab 64 a3 38 82 83 68 d3 03 ec 14 bf eb ac 2a 78 61 eb 78 42 9c 1c 95 3b 41 fa c6 10 43 64 9c 4b c4 09 4e 6c 14 b0 ed 26 a8 42 9a 74 39 ed 4a 58 3b 8f b4 e3 54 94 d8 10 e7 87 06 a5 85 a4 80 71 8d 3e b5 33 a7 18 4b c4 3d a2 d4 fe 26 ac 49 0d bd be 21 6e 74 98 30 ff 06 75 38 58 19 d8 e3 f4 93 4b bf 37 e8 cc 55 6d cf 69 c2 40 51 36 2a 70 c3 3f 25 49 b4 c5 71 99 05 b7 ec 26 e5 6e 4b e9 03 b7 c5 ce c2 3c 1f cf f6 49 ad b1 7a 3d 49 48 16 f9 ee 4e b8 4d 6b b5 93 b2 3a 3b f2 ee ca 5f	128	48.863	256	240
	b8 b9 81 61 f1 76 62 64 c6 0f 06 51 38 0d c8 72	192	46.798		
	5f 2f 7f 99 05 1e 0f 67 96 c4 4f da b8 2b 39 97	256	44.580		

5. CONCLUSION

In this paper, we introduce an approach aimed at enhancing the security of the AES block cipher through the modification of the Mixcolumn transformation utilizing efficient MDS matrices of sizes 4, 8, or 16. These matrices encompass three distinct types: Type-I circulant-like matrices, Hadamard matrices, and recursive matrices. Additionally, we elaborate on the function of the MDS matrix in augmenting the branch number of the block cipher's diffusion layer, consequently enhancing the cipher's security. We introduce an approach to derive a novel diffusion matrix for the modified AES, enabling the assessment of both the number of fixed points and the fixed point coefficient $D(A)$ within the modified AES diffusion layer. Moreover, we establish the branch number of the modified AES diffusion layer utilizing MDS matrices sized 8 and 16. Subsequently, we analyze the security, statistical benchmarks, and execution speed of the modified AES block ciphers resulting from these MDS matrices. The findings indicate a substantial enhancement in the security of the AES block cipher through our proposed method. In future research, we will study and improve other AES components such as Sboxes, and key schemes.

REFERENCES

[1] C. Boura, and M. N. Plasencia, "Impossible differential cryptanalysis," *Symmetric Cryptography, vol. 2: Cryptanalysis and Future Directions*. ISTE Ltd and John Wiley & Sons, Inc, 47, 2024.

[2] T. Beyne and V. Rijmen, "Differential cryptanalysis in the fixed-key model," in *Annual International Cryptology Conference Cham*. Springer Nature Switzerland, 2022, pp. 687–716.

[3] M. Matsui, "Linear cryptanalysis method for des cipher," *Advances in Cryptology—EUROCRYPT'93, LNCS 765*. Springer-Verlag, 1994, pp. 386–397.

[4] J. Shi, G. Liu, and C. Li, "SAT-based security evaluation for WARP against linear cryptanalysis," *IET Information Security*, vol. 2023, pp. 1-14, 2023. [Online]. Available: https://doi.org/10.1049/2023/5323380

[5] J. Daemen and V. Rijmen, *The Design of Rijndael: AES - The Advanced Encryption Standard*. Springer Berlin, Heidelberg, 2002. [Online]. Available: https://doi.org/10.1007/978-3-662-60769-5

[6] J. Daemen and V. Rijmen, "AES proposal: rijndael (version 2). nist aes website", pp 1-45, 1999. [Online]. Available: https://www.cs.miami.edu/home/burt/learning/Csc688.012/rijndael/rijndael_doc_V2.pdf

[7] L. Keliher, "Linear cryptanalysis of substitution-permutation networks," Queen's University, Kingston, Ontario, Canada, 2003.

[8] S. Samanta, "On the counting of involutory MDS matrices," *arXiv preprint arXiv:2310.00090*, 2023.

[9] J. Nakahara and E. Abrahao, "A new involutory mds matrix for the AES," *IJ Network Security*, vol. 9, no. 2, pp. 109–116, 2009.

[10] M. Sajadieh, M Dakhilalian, H. Mala, and B. Omoomi, "On construction of involutory MDS matrices from vandermonde matrices in GF(2q)," *Design, Codes and Cryptography*, vol. 64, no. 3, pp. 287-308, 2012.

[11] K. C. Gupta, S. K. Pandey, and Venkateswarlu, "Almost involutory recursive MDS diffusion layers," *Design, Codes and Cryptography*, vol. 87, pp. 609-626, 2018.

[12] T. T. Luong, N. N. Cuong, and B. D. Trinh, "4x4 recursive MDS matrices effective for implementation from reed-solomon code over GF(q) field," *International Conference on Modelling, Computation and Optimization in Information Systems and Management Sciences – MCO 2021,* 2021, pp 386–391.

[13] M. Liu and S. M. Sim, "Lightweight mds generalized circulant matrices," *Fast Software Encryption*. Springer, 2016, pp. 101-120.

[14] K.C. Gupta, and I.G. Ray, "On constructions of MDS matrices from circulant-like matrices for lightweight cryptography," Institution, Tech. Rep. ASU/2014/1, 2014.

[15] R. Oliynykov, I. Gorbenko, O. Kazymyrov, V. Ruzhentsev, O. Kuznetsov, Y. Gorbenko, and D. Kaidalov, *A new encryption standard of Ukraine: The Kalyna block cipher*. Cryptology ePrint Archive, Paper 2015/650, 2015. [Online]. Available:https://eprint.iacr.org/2015/650

[16] V. Dolmatov, "GOST R 34.12-2015: Block Cipher "Kuznyechik"" (No. rfc7801), 2016. [Online]. Available: https://www.rfc-editor.org/rfc/rfc7801

[17] H. T. Assafli and I. A. Hashim, "Generation and evaluation of a new time-dependent dynamic s-box algorithm for AES block cipher cryptosystems", 3*rd International Conference on Recent Innovations in Engineering (ICRIE 2020)*, Materials Science and Engineering 978 (2020) 012042 IOP Publishing, 2020. Doi:10.1088/1757-899X/978/1/012042

[18] J. Juremi, R. Mahmod, Z. A. Zukarnain, and Md. YasinS, "Modified AES s-box based on determinant matrix algorithm," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 7, no. 1, January 2017.

[19] A.H. Al-Wattar, R. Mahmod, Z.A. Zukarnain, and N. Udzir, "A new DNA based approach of generating key dependent Mixcolumns transformation," *International Journal of Computer Networks & Communications (IJCNC)*, vol. 7, no. 2, March 2015.

[20] I.A. Ismil, G.H.G. Edeen, S. Khattab, M.A. ElHamid, and I.M. Bahtity, "Performance examination of AES encrytion algorithm with constant and dynamic rotation," *International Journal of Reviews in Computing*, vol. 12, December 2012.

[21] F.Y. Asian, M.T. Sakalli, B. Asian, and S. Bulut, "A new involutory 4 x 4 MDS matrix for the AES-like block ciphers," *International Review on Computers and Software*, vol. 6, no. 1, pp. 96–103, 2011.

[22] R. Elumalai, and A.R. Reddy, "Improving diffusion power of aes rijndael with 8x8 MDS matrix," *International Journal of Scientific & Engineering Research*, vol. 2, pp. 1-5, 2011.

[23] J. Nakahara and E. Abrahao, "A new involutory MDS matrix for the AES," *Int. J. Netw. Secur.*, vol. 9, no. 2, pp. 109–116, 2009.

[24] A.M. Elhosary, N.H. Shaker, I.A.G. Farag, and A.E.R. Shehata, "Optimum dynamic diffusion of block cipher based on maximum distance separable matrices," *International Journal of Information and Network Security*, vol. 2, no. 4, p.327, 2013.

[25] T.T. Luong, "Proposing secure and efficient MDS matrices to improve the diffusion layer of the AES block cipher", *Proceedings of the 15th National Conference on Fundamental and Applied Information Technology Research (FAIR'2022)*, pp. 16-24, 2022.

[26] F.J. MacWilliams and N.J.A. Sloane, *The Theory of Error-Correcting Codes*. Elsevier, 1977.

[27] R. Elumalai and A.R. Reddy, "Improving diffusion power of AES rijndael with 8x8 MDS matrix," *International Journal of Scientific & Engineering Research*, vol. 2, pp. 1-5, 2011.

[28] T.T. Luong and N.N. Cuong, "Direct exponent and scalar multiplication transformation of MDS matrices: Some good cryptographic results for dynamic diffusion layers of block ciphers," *Journal of Computer Science and Cybernetics*, vol. 32, no. 1, pp. 1-17, 2016.

Received on December 19, 2023
Revised on February 28, 2024