# Construction of Lightweight MDS Matrices from Generalized Feistel Structures

Mahdi Sajadieh[1] and Mohsen Mousavi[2]

[1] Islamic Azad University, Isfahan, Iran, m.sajadieh@khuisf.ac.ir
[2] Malek Ashtar University, Isfahan, Iran, m.mousavi@mut-es.ac.ir

**Abstract.** This paper investigates the construction of lightweight MDS matrices with generalized Feistel structures (GFS). The approach developed by this paper consists in deriving MDS matrices from the product of several sparser ones. This can be seen as a generalization to several matrices of the recursive construction which derives MDS matrices as the powers of a single companion matrix. The first part of this paper gives some theoretical results on the iteration of GFS and the second part gives concrete instantiations. The results match the best known lightweight $4 \times 4$ MDS matrix and improve the best known $6 \times 6$ and $8 \times 8$ MDS matrices.
Based on GFS structure, we propose some types of sparse matrices that are called EGFS matrices. Then, by applying binary linear functions to several round of EGFS matrices, we propose lightweight $4 \times 4$, $6 \times 6$ and $8 \times 8$ MDS matrices which are implemented with 67, 158 and 272 XOR for 8-bit input, respectively. The major work of this paper is the design of an $8 \times 8$ MDS matrix with 272 XOR for 8-bit input, since the best known result is 392 XOR.

**Keywords:** Lightweight cryptography · MDS matrix · Generalized Feistel Structures

## 1 Introduction

There are several approaches to construct MDS matrices which can be applied as diffusion layers for block ciphers and hash functions. The first method is based on the algebraic structures such as Cauchy matrix [You97, Gup13]. The next efficient method to be used in construction of MDS matrices is based on the recursive matrices [Guo11, Saj12, Aug14, Ber13]. The first two approaches are called local optimization, since these methods focus on the implementation cost of entries of an MDS matrix [Sar16, Sim15].

Recently, two new approaches are proposed that are called global optimization [Kra17, Bei16, Duv18]. The proposed method in [Kra17] is an application of heuristics algorithms to obtain a suitable implementation of previously known MDS matrices. The introduced technique in [Bei16, Duv18] is a type of search over formal matrices independently of binary linear functions ($\mathbf{L}$) and to instantiate $\mathbf{L}$ later. In fact, the work of [Bei16, Duv18] is not an application of heuristics algorithms to obtain an efficient implementation of previously known MDS matrices, but a search for new lightweight MDS matrices starting from the implementation.

The space of matrix explored in this paper is a subspace of the space explored in [Duv18]. Although, the presented technique in this paper is exactly the same as [Duv18], the class of construction considered is different. In fact, the advantage of this work compared to [Duv18] is that the smaller search space can be used with larger dimensions.

In recursive or LFSR-based approach, we consider an $n \times n$ companion matrix $\mathbf{A}$ such that the entries of the last row of $\mathbf{A}$ are no-zero elements over a field $\mathbb{F}$. Then we check whether the $n$the power of $\mathbf{A}$, denoted with $\mathbf{A}^n$, is an MDS matrix over $\mathbb{F}$. The limitation

to the recursive approach is that all entries of the last row of $\mathbf{A}$ must be non-zero. In other words, the number of non-zero entries of $\mathbf{A}$ must be at least $2n-1$. Applying sparse matrices based on the Feistel structures is one of the best solutions for the mentioned limitation [Shi11, Wu12]. Actually, by applying Feistel structures, we can construct an $n \times n$ sparse matrix $\mathbf{B}$ so that $\mathbf{B}^n$ is an MDS matrix and also the number of non-zero entries of $\mathbf{B}$ is less than $2n-1$. The next limitation of construction an $n \times n$ MDS matrix based on the recursive approach or Feistel structures is that the $n$th power of a matrix is used as an MDS matrix. In fact, it is not possible to select an $n \times n$ companion matrix $\mathbf{A}$ such that all entries of $\mathbf{A}^k$ are non-zero provided that $k < n$. In addition, all known lightweight $n \times n$ MDS matrices which are derived from the Feistel structure are constructed from the $n$th power of $n \times n$ sparse matrices [Toh18].

In this paper, using binary linear functions ($\mathbf{L}$) and applying generalized Feistel structures we select $n \times n$ sparse matrices $\mathbf{A}_i$ with $1 \leq i \leq k$ such that $\mathbf{A}_i$'s satisfy the following conditions. First $\mathbf{A}_i$'s have the same structure with respect to the location of their zero entries. Then $\mathbf{A}_i$'s are non-singular matrices over $\mathbb{F}_2[\mathbf{L}]$. Moreover, the multiplication of $\mathbf{A}_i$'s is an MDS matrix over $\mathbb{F}_2[\mathbf{L}]$. The last and the main condition is that $\mathbf{A}_i$'s can be implemented with the low cost by terminology of XOR counts. Table 1 are provided for comparison our results with the best known results.

**Contribution of this work** This paper follows a list of recent papers to design new MDS matrices with low implementation costs. In particular, this paper presented several new matrices having lower XOR cost than previous results. While for $4 \times 4$ matrices, the results match the best known lightweight $4 \times 4$ MDS matrix [Duv18] the results for $6 \times 6$ are slightly better and for $8 \times 8$ are substantially better. Recently, $8 \times 8$ MDS matrices are applied in some block ciphers and hash functions such as LED, versions of PHOTON with 256 block length and Grøstl.

One of the Feistel-based structures that is used in this paper is GFS structure. Actually, using binary linear functions over GFS structure we propose two new results on lightweight MDS matrices. First, we propose an $6 \times 6$ MDS matrix which is implemented with 158 XOR for 8-bit input such that the proposed matrix is constructed from seven binary linear functions. The second new result is the implementation of an $8 \times 8$ MDS matrix with 272 XOR for 8-bit input. Moreover, the proposed $8 \times 8$ MDS matrix is constructed from the multiplication of seven matrices $\mathbf{A}_i$ with $1 \leq i \leq 7$ such that $\mathbf{A}_i$'s are derived from GFS structure. In addition, the number of binary linear functions that are applied in construction of $\mathbf{A}_i$'s is 16. In other words, the proposed $8 \times 8$ MDS matrix is not only has low implementation cost by hardware terminology, but also is suitable from software perspective. Furthermore, the depth of the proposed $8 \times 8$ MDS matrix is 9.

Moreover, we propose an $8 \times 8$ MDS matrix $\mathbf{B}$ for 8-bit input such that $\mathbf{B}$ and its inverse are implemented with 280 XOR over finite field. In addition, by applying binary linear functions, the proposed matrix $\mathbf{B}$ is implemented with $28 \times 2^k + 60$ XOR for $2^k$-bit input with $k > 3$.

**Outline of this paper** The rest of paper is organized as follows. Definitions and notations are given in Section 2. In Section 3, it is motivated why primitive matrices are used in this paper. Definition of primitive GFS matrices is provided in Section 4. Moreover, by using primitive GFS matrices a probabilistic algorithm for construction lightweight MDS matrices are proposed in Section 4. In Section 5, by applying binary linear functions over primitive GFS matrices, $4 \times 4$ and $8 \times 8$ lightweight MDS matrices are proposed. Moreover in Section 5, the proposed MDS matrices and their inverse are implemented with 68 and 280 XOR. An extension of primitive GFS matrices, called EGFS matrices, is given in Section 6. Furthermore, the best result of this paper by applying binary linear functions over EGFS matrices are provided in Section 6. Finally, a conclusion is given in Section 7.

**Table 1:** Comparison our results with the best known results.

| Type | Input Size | Cost | Method | Depth | Reference |
|---|---|---|---|---|---|
| $4 \times 4$ Matrices | | | | | |
| $\mathbb{F}_2[\mathbf{L}]$ | 4-bit | 41 | Lightweight Cir. | 3 | [Duv18] |
| $\mathbb{F}_2[\mathbf{L}]$ | 4-bit | 37 | GFS St. | 4 | [Duv18] |
| $\mathbb{F}_{2^4}$ | 4-bit | 36 | Heuristics Al. | — | [Kra17] |
| $\mathbb{F}_2[\mathbf{L}]$ | 4-bit | 36 | GFS St. | 6 | Subsection 5.1 |
| $\mathbb{F}_2[\mathbf{L}]$ | 4-bit | 35 | GFS St. | 5 | [Duv18] |
| $\mathbb{F}_2[\mathbf{L}]$ | 4-bit | 35 | GFS St. | 5 | Subsection 6.1 |
| $\mathbb{F}_2[\mathbf{L}]$ | 8-bit | 77 | Lightweight Cir. | 3 | [Duv18] |
| $\mathbb{F}_{2^8}$ | 8-bit | 72 | Subfield | — | [Kra17] |
| $\mathbb{F}_{2^8}$ | 8-bit | 70 | GFS St. | 5 | Subsection 6.1 |
| $\mathbb{F}_2[\mathbf{L}]$ | 8-bit | 69 | GFS St. | 4 | [Duv18] |
| $\mathbb{F}_2[\mathbf{L}]$ | 8-bit | 68 | GFS St. | 6 | Subsection 5.1 |
| $\mathbb{F}_2[\mathbf{L}]$ | 8-bit | 67 | GFS St. | 5 | [Duv18] |
| $\mathbb{F}_2[\mathbf{L}]$ | 8-bit | 67 | GFS St. | 5 | Subsection 6.1 |
| $6 \times 6$ Matrices | | | | | |
| $\mathbb{F}_{2^8}/\texttt{0x1C3}$ | 8-bit | 186 | DSI St. | — | [Toh18] |
| $\mathbb{F}_{2^8}$ | 8-bit | **158** | GFS St. | **8** | Subsection 6.2 |
| $\mathbb{F}_2[\mathbf{L}]$ | 8-bit | **158** | GFS St. | **8** | Subsection 6.2 |
| $8 \times 8$ Matrices | | | | | |
| $\mathbb{F}_{2^8}$ | 8-bit | 392 | Subfield | — | [Kra17] |
| $\mathbb{F}_{2^8}/\texttt{0x1A3}$ | 8-bit | **280** | GFS St. | 11 | Subsubsection 5.2 |
| $\mathbb{F}_2[\mathbf{L}]$ | 8-bit | **280** | GFS St. | 11 | Subsubsection 5.2 |
| $\mathbb{F}_2[\mathbf{L}]$ | 8-bit | **272** | GFS St. | **9** | Subsection 6.3 |

## 2 Definitions and Notations

Let $\mathbf{A}$ be an $n \times n$ matrix over a field $\mathbb{F}_q$, the finite field with $q$ elements. $\mathbf{A}$ is called MDS over $\mathbb{F}_q$ if any square submatrix of $\mathbf{A}$ is nonsingular over $\mathbb{F}_q$ [Bla99]. Moreover, a finite field with characteristic 2 is denoted with $\mathbb{F}_{2^q}$ for some $q$. Furthermore, we present a finite field $\mathbb{F}_{2^q}$ by hexadecimal representation. For instance, $\mathbb{F}_{2^8}/\texttt{0x18D}$ is the finite field $\mathbb{F}_{2^8}$ which is constructed from the primitive polynomial $\mathbf{f} = x^8 + x^7 + x^3 + x^2 + 1$. For simplicity, we use non-zero positions in each row of a binary matrix as a representation of the matrix. As an example, $[[1,2,4],[1,3],[2,4],[3,4]]$ is applied for $\mathbf{A} = \begin{pmatrix} 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{pmatrix}$.

We denote a matrix with $(a_{i,j})$ where $a_{i,j}$ is the $(i,j)$th entry of the matrix. Consider an $n \times n$ matrix $\mathbf{A} = (a_{i,j})$ with $1 \leq i, j \leq n$ over $\mathbb{R}$, the field of real numbers. Then $\mathbf{A}$ is called a positive matrix over $\mathbb{R}$ provided that $a_{i,j} > 0$ for any $1 \leq i, j \leq n$.

**Definition 1** ([Hor13])**.** Consider an $n \times n$ non-negative matrix $\mathbf{A}$ over $\mathbb{R}$. The matrix $\mathbf{A}$ is called a primitive matrix over $\mathbb{R}$ if $\mathbf{A}^k$ is a positive matrix, denoted $\mathbf{A}^k > 0$, for some integer $k \geq 1$. The primitive order $\mathbf{A}$ is the minimum number $k$ which satisfies $\mathbf{A}^k > 0$ and the matrix $\mathbf{A}$ is called an $k$-primitive matrix over $\mathbb{R}$.

Moreover, an $n \times n$ non-negative matrix $\mathbf{A}$ is called an $\infty$-primitive matrix over $\mathbb{R}$ if there is no an integer number $k \geq 1$ such that $\mathbf{A}^k > 0$.

In this paper, the symbol $\mathbb{F}_2[\mathbf{L}]$ is considered as a set of all finite polynomials in the following form $\sum_{i=1}^{n} b_i \mathbf{L}^{t_i}$ where $b_i \in \mathbb{F}_2$ and $t_i$'s are integer numbers. Consider an $n \times n$ matrix $\mathbf{A}$ over $\mathbb{F}_2[\mathbf{L}]$. Then $\mathbf{A} = (a_{i,j})$ is called a positive matrix over $\mathbb{F}_2[\mathbf{L}]$ if $a_{i,j} \neq 0$ for

any $1 \leq i, j \leq n$. Moreover, $\mathbf{A}$ is called an MDS matrix over $\mathbb{F}_2[\mathbf{L}]$ if determinant of all square submatrices of $\mathbf{A}$ are non-zero over $\mathbb{F}_2[\mathbf{L}]$.

Assume that $\mathbf{r}$ is a set over $\mathbb{F}_2[\mathbf{L}]$. Then the set of all prime factors of $\mathbf{r}$ is called the base set of $\mathbf{r}$. For instance, consider the set $\mathbf{r} = \{\mathbf{L}, \mathbf{L}^2, \mathbf{L}+1, \mathbf{L}^2+1, (\mathbf{L}^2+\mathbf{L}+1)^2, \mathbf{L}^6+\mathbf{L}^2+1\}$. Then the base set of $\mathbf{r}$ is $\widetilde{\mathbf{r}} = \{\mathbf{L}, \mathbf{L}+1, \mathbf{L}^2+\mathbf{L}+1, \mathbf{L}^3+\mathbf{L}+1\}$, since we have $\mathbf{L}^2+1 = (\mathbf{L}+1)^2$ and $\mathbf{L}^6+\mathbf{L}^2+1 = (\mathbf{L}^3+\mathbf{L}+1)^2$ over $\mathbb{F}_2[\mathbf{L}]$. Now consider an $n \times n$ non-singular matrix $\mathbf{A}$ over $\mathbb{F}_2$. If $\mathbf{A}$, $\mathbf{A}+\mathbf{I}_n$, $\mathbf{A}^2+\mathbf{A}+\mathbf{I}_n$ and $\mathbf{A}^3+\mathbf{A}+\mathbf{I}_n$ are non-singular matrices over $\mathbb{F}_2$, then we say the elements of $\widetilde{\mathbf{r}}$ are non-singular matrices over $\mathbb{F}_2$ by applying $\mathbf{A}$. Notice that if by using an $n \times n$ matrix $\mathbf{A}$ the elements of $\widetilde{\mathbf{r}}$ are non-singular matrices over $\mathbb{F}_2$, then it can be verified that the elements of $\mathbf{r}$ are non-singular matrices over $\mathbb{F}_2$ by applying $\mathbf{A}$. The last concept used in this paper is the depth of a circuit that determines the maximum number of gates in each path from any source to the sink [Duv18].

## 3 Relation between Primitive Matrices and Search Space

In this section, the role of primitive matrices for construction of proposed lightweight MDS matrices are explained. Consider $\mathbf{A}$ is an $n \times n$ non-negative matrix over $\mathbb{R}$. It is easy to see that if $\mathbf{A}$ is an $\infty$-primitive matrix over $\mathbb{R}$ then $\mathbf{A}$ can not be a primitive matrix over $\mathbb{F}_2[\mathbf{L}]$. Next, we show that if $\mathbf{A}$ is an $k$-primitive matrix over $\mathbb{R}$, then $\mathbf{A}$ can be an $\infty$-primitive over $\mathbb{F}_2[\mathbf{L}]$ or $\mathbf{A}$ be an $k'$-primitive matrix over $\mathbb{F}_2[\mathbf{L}]$ such that $k' \geq k$. In other words, the characteristic 2 in $\mathbb{F}_2[\mathbf{L}]$, puts limitation on order of primitive matrices.

**Example 1.** Consider the following three matrices

$$\mathbf{A}_1 = \begin{pmatrix} \mathbf{L} & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \mathbf{L} & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}, \quad \mathbf{A}_2 = \begin{pmatrix} \mathbf{L} & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}, \quad \mathbf{A}_3 = \begin{pmatrix} \mathbf{L} & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ \mathbf{L} & 0 & 0 & 0 \end{pmatrix} \tag{1}$$

Assume that $\mathbf{L}$ is a positive integer in $\mathbb{R}$. Then it can be checked that $\mathbf{A}_1$, $\mathbf{A}_2$ and $\mathbf{A}_3$ are 4-primitive matrices over $\mathbb{R}$. For instance, $\mathbf{A}_1^4$ over $\mathbb{R}$ is in the following form.

$$\mathbf{A}_1^4 = \begin{pmatrix} \mathbf{L}^4+1 & \mathbf{L}^3 & 3\,\mathbf{L}^2 & 2\,\mathbf{L} \\ 2\,\mathbf{L} & 1 & \mathbf{L}^3 & \mathbf{L}^2 \\ 3\,\mathbf{L}^2 & 2\,\mathbf{L} & \mathbf{L}^4+1 & \mathbf{L}^3 \\ \mathbf{L}^3 & \mathbf{L}^2 & 2\,\mathbf{L} & 1 \end{pmatrix}.$$

In addition, there is no integer number $1 \leq k \leq 3$ such that $\mathbf{A}_1^k$ be a positive matrix over $\mathbb{R}$. Now consider $\mathbf{A}_1$, $\mathbf{A}_2$ and $\mathbf{A}_3$ over $\mathbb{F}_2[\mathbf{L}]$.

First of all, we prove $\mathbf{A}_1$ is an $\infty$-primitive matrix over $\mathbb{F}_2[\mathbf{L}]$. The characteristic polynomial of $\mathbf{A}_1$ over $\mathbb{F}_2[\mathbf{L}]$ is $x^4+\mathbf{L}^2x^2+1$. Consider the equation $x^k = (x^4+\mathbf{L}^2x^2+1)h(x)+r(x)$ where $r(x)$ is a polynomial of degree less than 4 over $\mathbb{F}_2[\mathbf{L}]$. Therefore, we get $\mathbf{A}_1^k = r(\mathbf{A}_1)$, since $\mathbf{A}_1$ satisfies its own characteristic equation [Hor13]. It can be verified that $r(x) = a_1 + a_2x^2$ when $k$ is an even number and $r(x) = b_1x + b_2x^3$ when $k$ is an odd number where $a_i$ and $b_i$ with $1 \leq i \leq 2$, are in $\mathbb{F}_2[\mathbf{L}]$. Hence, $\mathbf{A}_1^k$ is a linear combination of ($\mathbf{I}_4$ and $\mathbf{A}_1^2$) or ($\mathbf{A}_1$ and $\mathbf{A}_1^3$) where $\mathbf{I}_4$ is the identity matrix of order 4. Moreover, $\mathbf{A}_1$, $\mathbf{A}_1^2$ and $\mathbf{A}_1^3$ are not positive matrices over $\mathbb{F}_2[\mathbf{L}]$. Furthermore, it can be checked that ($\mathbf{I}_4$ and $\mathbf{A}_1^2$) and ($\mathbf{A}_1$ and $\mathbf{A}_1^3$) have zero entries in the same positions. Therefore, for any positive integer $k$, $\mathbf{A}_1^k$ has at least one zero entry which results in $\mathbf{A}_1$ is an $\infty$-primitive matrix over $\mathbb{F}_2[\mathbf{L}]$. In addition, $\mathbf{A}_2$ and $\mathbf{A}_3$ are 7-primitive and 4-primitive matrices over $\mathbb{F}_2[\mathbf{L}]$.

Suppose that $\mathbf{A} = (a_{i,j})$ and $\mathbf{B} = (b_{i,j})$ with $1 \leq i, j \leq n$ are two $n \times n$ sparse matrices over $\mathbb{R}$. In this paper, $\mathbf{A}$ and $\mathbf{B}$ are called with the same structure provided that $a_{i,j} = 0$ if and only if $b_{i,j} = 0$. For instance, $4 \times 4$ sparse matrices $\mathbf{A}_1$, $\mathbf{A}_2$ and $\mathbf{A}_3$, given in Example 1, are with the same structure.

Assume that $\mathbf{A}_i$ with $1 \leq i \leq k$ are $n \times n$ sparse matrices over $\mathbb{R}$. Now based on the primitivity and structures of $\mathbf{A}_i$'s the following five cases are considered.

**Case** 1: $\mathbf{A}_i$'s are $\infty$-primitive matrices over $\mathbb{R}$ and are with the same structure.

**Case** 2: $\mathbf{A}_i$'s are $\infty$-primitive matrices over $\mathbb{R}$ and are not with the same structure.

**Case** 3: $\mathbf{A}_i$'s are primitive matrices over $\mathbb{R}$ and are with the same structure.

**Case** 4: $\mathbf{A}_i$'s are primitive matrices over $\mathbb{R}$ and are not with the same structure.

**Case** 5: Let $\mathbf{I} = \{j_1, j_2, \cdots, j_m\}$ with $1 \leq m < k$ be a subset of the set $\{1, 2, \cdots, k\}$. Consider $\mathbf{A}_{j_t}$ with $1 \leq t \leq m$ are primitive matrices over $\mathbb{R}$ and $\mathbf{A}_i$ for $i \notin \mathbf{I}$ are not primitive matrices over $\mathbb{R}$ (the structures of $\mathbf{A}_i$'s can be chosen arbitrarily).

By considering the given five cases, suppose $\mathbf{B}$ is the multiplication of $\mathbf{A}_i$'s, denoted with $\mathbf{B} = \prod_{i=1}^{k} \mathbf{A}_i$. It follows from graph theory concepts that by assuming the case 1 the matrix $\mathbf{B}$ is not a primitive matrix over $\mathbb{F}_2[\mathbf{L}]$. Moreover, $\mathbf{B}$ is possibly a primitive matrix over $\mathbb{F}_2[\mathbf{L}]$ by considering the assumptions of 2,3,4 and 5 cases and hence the matrix $\mathbf{B}$ possibly can be an MDS matrix over $\mathbb{F}_2[\mathbf{L}]$. Therefore, to construct MDS matrices over $\mathbb{F}_2[\mathbf{L}]$, by applying search on sparse matrices, 2,3,4 and 5 cases can be used.

Performing a search using the third case has less complexity than 2,4 and 5 cases, since $\mathbf{A}_i$'s are with the same structure. In fact, firstly we obtain an $n \times n$ $k$-primitive matrix $\mathbf{C}$ over $\mathbb{R}$. Then we choose matrices $\mathbf{A}_i$ with $1 \leq i \leq k'$ over $\mathbb{F}_2[\mathbf{L}]$ provided that $k' \geq k$ and $\mathbf{A}_i$'s are with the same structure as $\mathbf{C}$. Next, we check whether $\mathbf{B} = \prod_{i=1}^{k'} \mathbf{A}_i$ is an MDS matrix over $\mathbb{F}_2[\mathbf{L}]$. Therefore, the main reason for using primitive matrices with the same structure is the issue of reducing the search space.

**Example 2.** Consider the following $4 \times 4$ sparse matrices over $\mathbb{F}_2[\mathbf{L}]$.

$$\widetilde{\mathbf{A}}_1 = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \widetilde{\mathbf{A}}_2 = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & \mathbf{L} & 0 & 0 \end{pmatrix}, \widetilde{\mathbf{A}}_3 = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \mathbf{L} \\ 0 & 0 & 0 & 1 \\ \mathbf{L} & \mathbf{L} & 0 & 0 \end{pmatrix}, \widetilde{\mathbf{A}}_4 = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \end{pmatrix}.$$

The matrices $\widetilde{\mathbf{A}}_i$ with $1 \leq i \leq 4$ are extracted from Appendix C.2, Figure 8 in [Duv18]. It is shown in [Duv18] the multiplication of $\widetilde{\mathbf{A}}_i$'s, denoted with $\widetilde{\mathbf{B}} = \prod_{i=1}^{4} \widetilde{\mathbf{A}}_i$, is an MDS matrix over $\mathbb{F}_2[\mathbf{L}]$. Moreover, it can be checked that $\widetilde{\mathbf{B}}$ is obtained from the Case 5, since $\widetilde{\mathbf{A}}_2$, $\widetilde{\mathbf{A}}_3$ and $\widetilde{\mathbf{A}}_4$ are primitive matrices over $\mathbb{R}$ and $\widetilde{\mathbf{A}}_1$ is an $\infty$-primitive matrix over $\mathbb{R}$.

Consider the matrix $\mathbf{A}_1$ in Example 1. It is observed that $\mathbf{A}_1$ is an 4-primitive matrix over $\mathbb{R}$. Now we select matrices $\widehat{\mathbf{A}}_i$ with $1 \leq i \leq 4$ provided that $\widehat{\mathbf{A}}_i$'s are with the same structure as $\mathbf{A}_1$. In addition, the multiplication of $\widehat{\mathbf{A}}_i$'s, denoted with $\widehat{\mathbf{B}} = \prod_{i=1}^{4} \widehat{\mathbf{A}}_i$, be an MDS matrix over $\mathbb{F}_2[\mathbf{L}]$. The following matrices are derived from a simple search.

$$\widehat{\mathbf{A}}_1 = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}, \widehat{\mathbf{A}}_2 = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \mathbf{L} & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}, \widehat{\mathbf{A}}_3 = \begin{pmatrix} \mathbf{L} & \mathbf{L} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \mathbf{L} & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}, \widehat{\mathbf{A}}_4 = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}.$$

Furthermore, by similar structural properties of $\widetilde{\mathbf{A}}_i$ and $\widehat{\mathbf{A}}_i$ with $1 \leq i \leq 4$ we conclude that $\widetilde{\mathbf{B}}$ and $\widehat{\mathbf{B}}$ can be implemented with the same XOR from hardware perspective.

In the next section, according to the technique given in [Duv18] and applying primitive matrices, a probabilistic algorithm for construction of lightweight MDS matrices are proposed. The output of the proposed algorithm are $n \times n$ sparse matrices $\mathbf{A}_i$ with $1 \leq i \leq u$ over $\mathbb{F}_2[\mathbf{L}]$ with the following conditions. First, $u \leq n$ and $\mathbf{A}_i$'s are with the same structure. Second, $\mathbf{A}_i$'s are $k$-primitive matrices over $\mathbb{R}$ such that $k \leq u$. Third, the multiplication of $\mathbf{A}_i$'s is an MDS matrix over $\mathbb{F}_2[\mathbf{L}]$. The last and foremost condition is that using binary linear functions the implementation cost of $\mathbf{A}_i$'s is minimal.

In other words, the proposed randomized algorithm is an algorithm for decomposition of an MDS matrix into sparse matrices provided that these sparse matrices first have the same and simple structures, then are non-singular matrices over the base field, and eventually are implemented with minimal XOR.

# 4 Primitive GFS Matrices

There are two reasons why GFS structure is used in this paper. The first and most important reason is this fact the inverse of GFS structure is easy to compute as well. The second one is that combining primitive matrices and GFS structure reduces search space.

In this section, based on the concept of GFS structure a type of primitive sparse matrix is proposed which is called primitive GFS matrix. In Section 5 by applying primitive GFS matrices, $4 \times 4$ and $8 \times 8$ MDS matrices are proposed such that the implementation cost of these matrices are 68 and 280 XOR for 8-bit input , respectively. Consider the following $2 \times 2$ block-matrices.

$$\mathbf{c}_1^{(m)} = \begin{pmatrix} \mathbf{L} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}, \quad \mathbf{c}_2^{(m)} = \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{1} & \mathbf{0} \end{pmatrix}, \quad \mathbf{z}^{(m)} = \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}. \tag{2}$$

where $\mathbf{L}$ is an $m \times m$ non-singular matrix over $\mathbb{F}_2$. In addition, $\mathbf{1}$ and $\mathbf{0}$ are $m \times m$ identity and zero matrices, receptively. In Definition (2), by applying $\mathbf{c}_1^{(m)}$, $\mathbf{c}_2^{(m)}$ and $\mathbf{z}^{(m)}$ we propose primitive GFS matrices.

**Definition 2** (Primitive GFS Matrices)**.** Consider $\mathbf{p}_1 = \{a_1, a_2, \cdots, a_n\}$ and $\mathbf{p}_2 = \{b_1, b_2, \cdots, b_n\}$ are two permutations of integer numbers from 1 to $n$ provided that $a_i \neq b_i$ for any $1 \leq i \leq n$. Consider $2 \times 2$ block-matrices $\mathbf{c}_1^{(m)}$, $\mathbf{c}_2^{(m)}$ and $\mathbf{z}^{(m)}$ that are given in (2).

Suppose that the $i$th row of an $n \times n$ block-matrix $\mathbf{S} = (s_{i,j})$ with $1 \leq i, j \leq n$, based on the two permutations $\mathbf{p}_1$ and $\mathbf{p}_2$, is filled in the following form.

$$s_{i,j} = \begin{cases} \mathbf{c}_1^{(m)} & j = a_i, \\ \mathbf{c}_2^{(m)} & j = b_i, \\ \mathbf{z}^{(m)} & j \notin \{a_i, b_i\}. \end{cases}$$

The block-matrix $\mathbf{S}$ is called primitive GFS matrix if $\mathbf{S}$ be a primitive matrix over $\mathbb{R}$.

First of all notice that in Definition 2 to check whether $\mathbf{S}$ is a primitive matrix over $\mathbb{R}$, we assume that $\mathbf{c}_1^{(m)}$, $\mathbf{c}_2^{(m)}$ and $\mathbf{z}^{(m)}$ are $2 \times 2$ matrices over $\mathbb{R}$ and $\mathbf{L}$ is a positive integer. Moreover, for simplicity the block-matrix $\mathbf{S}$ is denoted with $\mathbf{S}(n, \mathbf{L}_{(m)}, [\mathbf{p}_1, \mathbf{p}_2])$. Furthermore, it follows from Definition 2 that a primitive GFS matrix can be an $\infty$-primitive matrix over $\mathbb{F}_2[\mathbf{L}]$. In addition, primitive GFS matrices are non-singular over $\mathbb{F}_2[\mathbf{L}]$, since it is easy to prove $\det(\mathbf{S}) = 1$ over $\mathbb{F}_2[\mathbf{L}]$. In Example 3, it is observed that why the block-matrix $\mathbf{S}$ in Definition 2 should be a primitive matrix over $\mathbb{R}$ and not ask for $\mathbf{S}$ to be primitive over $\mathbb{F}_2[\mathbf{L}]$.

**Example 3.** Suppose that in Definition 2, we assumed $\mathbf{S}$ be a primitive matrix over $\mathbb{F}_2[\mathbf{L}]$. Consider the following $n \times n$ block-matrix that is constructed from two permutations $\mathbf{p}_1 = \{1, 2, \cdots, n\}$ and $\mathbf{p}_2 = \{2, 3, \cdots, n, 1\}$.

$$\mathbf{S}(n, \mathbf{1}_{(m)}, [\mathbf{p}_1, \mathbf{p}_2]) = \left(\begin{array}{cc|cc|cc|cc} \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{1} \\ \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{array}\right).$$

For simplicity in representation set $\mathbf{A} = \mathbf{S}(n, \mathbf{1}_{(m)}, [\mathbf{p}_1, \mathbf{p}_2])$. In the rest, we prove $\mathbf{A}$ is an $\infty$-primitive matrix over $\mathbb{F}_2[\mathbf{L}]$. Suppose $\mathbf{A}$ is a primitive matrix over $\mathbb{F}_2[\mathbf{L}]$ which implies that there is a positive integer $k$ such that $\mathbf{A}^k > 0$ over $\mathbb{F}_2[\mathbf{L}]$. Hence, all entries of $\mathbf{A}^k$ are equal to 1. But it is in contradiction to this fact that $\mathbf{A}$ is a non-singular matrix over $\mathbb{F}_2[\mathbf{L}]$ but $\mathbf{A}^k$ is a singular matrix over $\mathbb{F}_2[\mathbf{L}]$, since $\mathbf{A}^k$ has two equal rows. Therefore, $\mathbf{S}$ in Definition 2 should be asked to be a primitive matrix over $\mathbb{R}$ until we can apply lightweight block-matrices such as $\mathbf{A}$ in the proposed constructions.

It follows from Appendix A that the order of a primitive GFS matrix $\mathbf{S}(n, \mathbf{L}_{(m)}, [\mathbf{p}_1, \mathbf{p}_2])$ depends on two permutations $\mathbf{p}_1$ and $\mathbf{p}_2$. Moreover, the implementation cost of proposed MDS matrices in Section 5, is directly related to the number of iterations required to reach full diffusion. Therefore, for construction of lightweight MDS matrices, the primitive GFS matrices with minimum order should be used. In Example 4, some types of primitive GFS matrices are made, which are used in Section 5 to construct lightweight $4 \times 4$ and $8 \times 8$ MDS matrices with the implementation cost 68 and 280 XOR for 8-bit input , respectively.

**Example 4.** For $n = 2$, the following primitive GFS matrix is used.

$$\mathbf{p}_1 = \{1, 2\}, \quad \mathbf{p}_2 = \{2, 1\}, \quad \mathbf{S}(2, \mathbf{L}_{(m)}, [\mathbf{p}_1, \mathbf{p}_2]) = \left(\begin{array}{cc|cc} \mathbf{L} & \mathbf{1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{L} & \mathbf{1} \\ \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{array}\right). \tag{3}$$

In Example 1, it is observed that $\mathbf{S}(2, \mathbf{L}_{(m)}, [\mathbf{p}_1, \mathbf{p}_2])$ is an 4-primitive matrix over $\mathbb{R}$.

For $n = 3$, there are 12 permutations such as $\mathbf{p}_1 = \{a_1, a_2, a_3\}$ and $\mathbf{p}_2 = \{b_1, b_2, b_3\}$ from 1 to 3 so that $a_i \neq b_i$ for any $1 \leq i \leq 3$. Consider the following case

$$\mathbf{p}_1 = \{1, 3, 2\}, \quad \mathbf{p}_2 = \{2, 1, 3\}, \quad \mathbf{S}(3, \mathbf{L}_{(m)}, [\mathbf{p}_1, \mathbf{p}_2]) = \left(\begin{array}{cc|cc|cc} \mathbf{L} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{L} & \mathbf{1} \\ \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{L} & \mathbf{1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} \end{array}\right) \tag{4}$$

The primitive GFS matrix $\mathbf{S}(3, \mathbf{L}_{(m)}, [\mathbf{p}_1, \mathbf{p}_2])$, is an 5-primitive matrix over $\mathbb{R}$.

For $n = 4$, there are 216 permutations $\mathbf{p}_1 = \{a_1, a_2, a_3, a_4\}$ and $\mathbf{p}_2 = \{b_1, b_2, b_3, b_4\}$ from 1 to 4 such that $a_i \neq b_i$ for $1 \leq i \leq 4$. These 216 permutations are divided into four cases, which are listed in Appendix A. It follows from Appendix A that the minimum order of a primitive GFS matrix such as $\mathbf{S}(4, \mathbf{L}_{(m)}, [\mathbf{p}_1, \mathbf{p}_2])$ is 6. Therefore, we select two permutations $\mathbf{p}_1$ and $\mathbf{p}_2$ provided that $\mathbf{S}(4, \mathbf{L}_{(m)}, [\mathbf{p}_1, \mathbf{p}_2])$ is an 6-primitive matrix over $\mathbb{R}$.

$$\mathbf{p}_1 = \{4, 3, 2, 1\}, \quad \mathbf{p}_2 = \{3, 2, 1, 4\}, \quad \mathbf{S}(4, \mathbf{L}_{(m)}, [\mathbf{p}_1, \mathbf{p}_2]) = \left(\begin{array}{cc|cc|cc|cc} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{L} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{L} & \mathbf{1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{L} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{L} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} \end{array}\right), \tag{5}$$

$$\widehat{\mathbf{p}}_1 = \{1, 3, 2, 4\}, \quad \widehat{\mathbf{p}}_2 = \{2, 1, 4, 3\}, \quad \mathbf{S}(4, \mathbf{L}_{(m)}, [\widehat{\mathbf{p}}_1, \widehat{\mathbf{p}}_2]) = \left(\begin{array}{cc|cc|cc|cc} \mathbf{L} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{L} & \mathbf{1} & \mathbf{0} & \mathbf{0} \\ \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{L} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{L} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{array}\right). \tag{6}$$

Two primitive GFS matrices such as $\mathbf{A}_1$ and $\mathbf{A}_2$ are considered with the same structure if $\mathbf{A}_1$ and $\mathbf{A}_2$ are constructed with the same permutations. In the rest, we propose a probabilistic algorithm for construction of lightweight MDS matrices based on the primitive GFS matrices with the same structure. Actually, Algorithm 1 is a randomized algorithm, since with the proposed approach there is an extensive search space to construct lightweight MDS matrices for large dimensions. For $m = 4, 8$ in Algorithm 1, the following forms are applied as $\mathbf{L}$ in Step 8 where $b_{i,j}$ with $1 \leq i, j \leq 4$ and $c_{i,j}$ with $1 \leq i, j \leq 8$ are binary numbers.

$$\begin{pmatrix} b_{1,1} & b_{1,2} & b_{1,3} & 1 \\ 1 & b_{2,2} & b_{2,3} & b_{2,4} \\ 0 & 1 & b_{3,3} & b_{3,4} \\ 0 & 0 & 1 & b_{4,4} \end{pmatrix}, \quad \begin{pmatrix} c_{1,1} & c_{1,2} & c_{1,3} & c_{1,4} & c_{1,5} & c_{1,6} & c_{1,7} & 1 \\ 1 & c_{2,2} & c_{2,3} & c_{2,4} & c_{2,5} & c_{2,6} & c_{2,7} & c_{2,8} \\ 0 & 1 & c_{3,3} & c_{3,4} & c_{3,5} & c_{3,6} & c_{3,7} & c_{3,8} \\ 0 & 0 & 1 & c_{4,4} & c_{4,5} & c_{4,6} & c_{4,7} & c_{4,8} \\ 0 & 0 & 0 & 1 & c_{5,5} & c_{5,6} & c_{5,7} & c_{5,8} \\ 0 & 0 & 0 & 0 & 1 & c_{6,6} & c_{6,7} & c_{6,8} \\ 0 & 0 & 0 & 0 & 0 & 1 & c_{7,7} & c_{7,8} \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & c_{8,8} \end{pmatrix}.$$

**Algorithm 1:** Construction of Lightweight MDS Matrices based on the Primitive GFS Matrices with the same Structure

**Input :** Three positive integer $n$, $m$ and $r$.
**Output :** An $2n \times 2n$ lightweight MDS matrix over $m$-bit input with $\leq r$ XOR.

1 Select two permutations $\mathbf{p}_1$ and $\mathbf{p}_2$ such that the order of primitive GFS matrix $\mathbf{C} = \mathbf{S}(n, \mathbf{1}_{(m)}, [\mathbf{p}_1, \mathbf{p}_2])$ is minimal over $\mathbb{R}$.
2 Consider $\mathbf{C}$ is an $k$-primitive matrix over $\mathbb{R}$ and set $u = k$.
3 Select primitive GFS matrices $\mathbf{A}_i = \mathbf{S}(n, \mathbf{f}^{(i)}_{(m)}, [\mathbf{p}_1, \mathbf{p}_2])$ with $1 \leq i \leq u$ such that $\mathbf{f}^{(i)} \in \{\mathbf{L}^{-2}, \mathbf{L}^{-1}, \mathbf{1}, \mathbf{L}, \mathbf{L}^{-2}\}$ and $\mathbf{A}_i$'s are constructed of minimal number of $\mathbf{L}$.
4 Construct $\mathbf{B} = \prod_{i=1}^{u} \mathbf{A}_i$ over $\mathbb{F}_2[\mathbf{L}]$.
5 *If* $\mathbf{B}$ is an MDS matrix over $\mathbb{F}_2[\mathbf{L}]$ *then* go in Step 8 *end*.
6 *If* all cases in Step 3 are considered *then* set $u = u + 1$.
7 *If* $\mathbf{B}$ is not an MDS matrix over $\mathbb{F}_2[\mathbf{L}]$ *then* go in Step 3 *end*.
8 Get the base set of subdeterminants of $\mathbf{B}$ over $\mathbb{F}_2[\mathbf{L}]$.
9 Obtain an $m \times m$ non-singular binary matrix $\mathbf{L}$ with the minimal implementation cost provided that elements of the base set are non-singular matrices over $\mathbb{F}_2$ by $\mathbf{L}$.
10 *If* Step 9 fails to obtain a binary matrix $\mathbf{L}$ *then* go in Step 3 *end*.
11 Obtain the implementation cost of $\mathbf{A}_i$'s, denoted $x$, with respect to the cost of $\mathbf{L}$.
12 *If* $x \leq r$ *then* go in Step 14 *end*.
13 *If* $u \leq 2n$ *then* go in Step 3 *else* go in Step 1 *end*.
14 Return $\mathbf{B}$ , $\mathbf{A}_i$'s and the non-singular binary matrix $\mathbf{L}$.

**Example 5.** The best implementation of lightweight $6 \times 6$ MDS matrices for 8-bit input is 186 XOR [Toh18]. Set $n = 3$, $m = 8$ and $r = 186$ as input into Algorithm 1. It follows from (4) that $\mathbf{C} = \mathbf{S}(3, \mathbf{1}_{(8)}, [\mathbf{p}_1, \mathbf{p}_2])$ is an 5-primitive matrices over $\mathbb{R}$ for two permutations $\mathbf{p}_1 = \{1, 3, 2\}$ and $\mathbf{p}_2 = \{2, 1, 3\}$. Using complete search, it can be verified that there are no primitive GFS matrices $\mathbf{A}_i = \mathbf{S}(3, \mathbf{f}^{(i)}_{(m)}, [\mathbf{p}_1, \mathbf{p}_2])$ with $1 \leq i \leq 5$ such that $\prod_{i=1}^{5} \mathbf{A}_i$ be an MDS matrix over $\mathbb{F}_2[\mathbf{L}]$ provided that $\mathbf{f}^{(i)} \in \{\mathbf{L}^{-2}, \mathbf{L}^{-1}, \mathbf{1}, \mathbf{L}, \mathbf{L}^{-2}\}$. Now consider the following primitive GFS matrices $\mathbf{A}_i$ with $1 \leq i \leq 6$.

$$\begin{array}{lll} \mathbf{A}_1 = \mathbf{R}(3, \mathbf{1}_{(m)}, [\mathbf{p}_1, \mathbf{p}_2]), & \mathbf{A}_2 = \mathbf{R}(3, \mathbf{L}_{(m)}, [\mathbf{p}_1, \mathbf{p}_2]), & \mathbf{A}_3 = \mathbf{R}(3, \mathbf{1}_{(m)}, [\mathbf{p}_1, \mathbf{p}_2]), \\ \mathbf{A}_4 = \mathbf{R}(3, \mathbf{1}_{(m)}, [\mathbf{p}_1, \mathbf{p}_2]), & \mathbf{A}_5 = \mathbf{R}(3, \mathbf{L}^2_{(m)}, [\mathbf{p}_1, \mathbf{p}_2]), & \mathbf{A}_6 = \mathbf{R}(3, \mathbf{1}_{(m)}, [\mathbf{p}_1, \mathbf{p}_2]). \end{array}$$

It can be checked that $\mathbf{B} = \prod_{i=1}^{6} \mathbf{A}_i$ is an MDS matrix over $\mathbb{F}_2[\mathbf{L}]$. Moreover, using a complete search, matrix $\mathbf{B}$ is an optimal result with respect to the number of binary linear functions ($\mathbf{A}_i$'s are constructed from nine $\mathbf{L}$).

$$\mathbf{B} = \begin{pmatrix} \mathbf{L}^3+\mathbf{1} & \mathbf{L}^3+\mathbf{1} & \mathbf{L}^3+\mathbf{L}^2+\mathbf{L}+\mathbf{1} & \mathbf{L}^3+\mathbf{L}^2+\mathbf{1} & \mathbf{1} & \mathbf{L} \\ \mathbf{L}^3+\mathbf{L}+\mathbf{1} & \mathbf{L}^3+\mathbf{1} & \mathbf{L}^2 & \mathbf{L}^2+\mathbf{L} & \mathbf{L}^3+\mathbf{1} & \mathbf{L}^3 \\ \mathbf{1} & \mathbf{L} & \mathbf{L}^3+\mathbf{1} & \mathbf{L}^3+\mathbf{1} & \mathbf{L}^3+\mathbf{L}^2+\mathbf{L}+\mathbf{1} & \mathbf{L}^3+\mathbf{L}^2+\mathbf{1} \\ \mathbf{L}^3+\mathbf{1} & \mathbf{L}^3 & \mathbf{L}^3+\mathbf{L}+\mathbf{1} & \mathbf{L}^3+\mathbf{1} & \mathbf{L}^2 & \mathbf{L}^2+\mathbf{L} \\ \mathbf{L}^3+\mathbf{L}^2+\mathbf{L}+\mathbf{1} & \mathbf{L}^3+\mathbf{L}^2+\mathbf{1} & \mathbf{1} & \mathbf{L} & \mathbf{L}^3+\mathbf{1} & \mathbf{L}^3+\mathbf{1} \\ \mathbf{L}^2 & \mathbf{L}^2+\mathbf{L} & \mathbf{L}^3+\mathbf{1} & \mathbf{L}^3 & \mathbf{L}^3+\mathbf{L}+\mathbf{1} & \mathbf{L}^3+\mathbf{1} \end{pmatrix}.$$

It is easy to verify that the number of subdeterminants of an $n \times n$ matrix is $\sum_{i=1}^{n} \binom{n}{i}\binom{n}{i} = \binom{2n}{n} - 1$. Therefore, $\mathbf{B}$ has $\binom{12}{6} - 1 = 923$ subdeterminants. The base set of subdeterminants of $\mathbf{B}$ is given in (7).

$$\begin{array}{l} \{L, L+1, L^2+L+1, L^3+L+1, L^3+L^2+1, L^4+L+1, L^4+L^3+1, L^4+L^3+L^2+L+1, \\ L^5+L^2+1, L^5+L^3+1, L^5+L^3+L^2+L+1, L^5+L^4+L^2+L+1, L^5+L^4+L^3+L+1, \\ L^5+L^4+L^3+L^2+1, L^6+L+1, L^6+L^3+1, L^6+L^4+L^2+L+1, L^6+L^4+L^3+L+1, \\ L^6+L^5+1, L^6+L^5+L^2+L+1, L^6+L^5+L^3+L^2+1, L^6+L^5+L^4+L+1, L^6+L^5+L^4+L^2+1, \\ L^7+L^3+L^2+L+1, L^7+L^5+L^3+L+1, L^7+L^6+1, L^7+L^6+L^4+L^2+1, L^7+L^6+L^5+L^4+1, \\ L^8+L^6+L^5+L^3+1, L^8+L^7+L^3+L+1, L^8+L^7+L^6+L^3+L^2+L+1\}. \end{array} \tag{7}$$

Assume that $\#\mathbf{L}$ is the symbol for XOR cost of $\mathbf{L}$ . Therefore, the implementation cost of $\mathbf{B}$ for $m$-bit input is computed as follows

$$\overbrace{(3m)}^{\#\mathbf{B}_1}+\overbrace{(3m+3(\#\mathbf{L})}^{\#\mathbf{B}_2}+\overbrace{(3m)}^{\#\mathbf{B}_3}+\overbrace{(3m)}^{\#\mathbf{B}_4}+\overbrace{(3m+3(\#\mathbf{L}^2)}^{\#\mathbf{B}_5}+\overbrace{(3m)}^{\#\mathbf{B}_6}=18m+3(\#\mathbf{L})+3(\#\mathbf{L}^2). \tag{8}$$

In the rest, we get $8\times 8$ non-singular matrices $\mathbf{L}$ over $\mathbb{F}_2$ such that the given elements in (7) are non-singular matrices over $\mathbb{F}_2$ by applying $\mathbf{L}$. For instance, the following non-singular binary matrices $\mathbf{L}_i$ with $1\leq i\leq 10$, are obtained.

$$\begin{array}{ll}
\mathbf{L}_1=[[1,8],[1],[2,7],[3],[4],[5],[6],[7]], & \mathbf{L}_2=[[1,8],[1],[2],[3],[4,7],[5],[6],[7]],\\
\mathbf{L}_3=[[3,8],[1],[2],[3],[4],[5,6],[6],[7]], & \mathbf{L}_4=[[5,8],[1],[2],[3],[4],[5],[6,7],[7]],\\
\mathbf{L}_5=[[8],[1,2],[2],[3,8],[4],[5],[6],[7]], & \mathbf{L}_6=[[8],[1,2],[2],[3],[4],[5,8],[6],[7]],\\
\mathbf{L}_7=[[8],[1,4],[2],[3],[4],[5],[6,7],[7]], & \mathbf{L}_8=[[8],[1,6],[2],[3],[4],[5],[6],[7,8]],\\
\mathbf{L}_9=[[8],[1],[2,3],[3],[4],[5,8],[6],[7]], & \mathbf{L}_{10}=[[8],[1],[2,5],[3],[4],[5],[6],[7,8]].
\end{array} \tag{9}$$

The implementation cost of $\mathbf{L}_i$'s, given in (9), is two XOR. In addition $\mathbf{L}_i^2$ with $1\leq i\leq 10$, can be implemented with four XOR. Therefore, it follows from (8) that the implementation cost of $\mathbf{B}$ is $18\times 8+3\times 2+3\times 4=162$ XOR. Moreover, $\mathbf{B}^{-1}$ is implemented with 162 XOR, since $\mathbf{B}^{-1}=\prod_{i=1}^{6}\mathbf{A}_{7-i}^{-1}$. Furthermore, consider an irreducible polynomial of degree 8 over $\mathbb{F}_2$ provided that this polynomial is not an element of (7) such as `0x11B`. Assume that $\alpha$ is a root of `0x11B`. Then by substitution $\alpha$ instead of $\mathbf{L}$, the matrix $\mathbf{B}$ is an MDS matrix over $\mathbb{F}_{2^8}/\texttt{0x11B}$.

# 5 Construction of Lightweight MDS Matrices by Applying Primitive GFS Matrices

In this section, by applying primitive GFS matrices, we propose lightweight $4\times 4$ and $8\times 8$ MDS matrices for 8 bit input. The proposed $4\times 4$ and $8\times 8$ MDS matrices are implemented with 68 and 280 XOR. Moreover, the given results in this section are optimal according to the terminology of XOR cost and the number of binary linear functions. In other words, it is not possible to achieve better results by using the proposed approach in Section 4. In fact, in order to obtain the results of this section, an exhaustive search has been applied to Algorithm 1. The point of the proposed $4\times 4$ and $8\times 8$ MDS matrices is the inverse of these matrices which can be implemented with 68 and 280 XOR, respectively.

## 5.1 Lightweight $4\times 4$ MDS Matrices from Primitive GFS Matrices

Consider the primitive GFS matrix $\mathbf{C}=\mathbf{R}(2,\mathbf{1}_{(m)},[\mathbf{p}_1,\mathbf{p}_2])$, over two permutations $\mathbf{p}_1=\{1,2\}$ and $\mathbf{p}_2=\{2,1\}$. It follows from (3) that $\mathbf{C}$ is an 4-primitive matrix over $\mathbb{R}$. In the rest, based on the matrix structure $\mathbf{C}$, the following primitive GFS matrices are used to construct a lightweight $4\times 4$ MDS matrix for $m$-bit input.

$$\begin{array}{ll}
\mathbf{A}_1=\mathbf{R}(2,\mathbf{1}_{(m)},[\mathbf{p}_1,\mathbf{p}_2]), & \mathbf{A}_2=\mathbf{R}(2,\mathbf{L}_{(m)},[\mathbf{p}_1,\mathbf{p}_2]),\\
\mathbf{A}_3=\mathbf{R}(2,\mathbf{L}_{(m)},[\mathbf{p}_1,\mathbf{p}_2]), & \mathbf{A}_4=\mathbf{R}(2,\mathbf{1}_{(m)},[\mathbf{p}_1,\mathbf{p}_2]).
\end{array}$$

It can be checked that $\mathbf{B}=\mathbf{A}_1\mathbf{A}_2\mathbf{A}_3\mathbf{A}_4$, given in (10), is an MDS matrix over $\mathbb{F}_2[\mathbf{L}]$.

$$\begin{aligned}
\mathbf{B} &= \begin{pmatrix}\mathbf{1}&\mathbf{1}&\mathbf{0}&\mathbf{0}\\ \mathbf{0}&\mathbf{0}&\mathbf{1}&\mathbf{0}\\ \mathbf{0}&\mathbf{0}&\mathbf{1}&\mathbf{1}\\ \mathbf{1}&\mathbf{0}&\mathbf{0}&\mathbf{0}\end{pmatrix}\begin{pmatrix}\mathbf{L}&\mathbf{1}&\mathbf{0}&\mathbf{0}\\ \mathbf{0}&\mathbf{0}&\mathbf{1}&\mathbf{0}\\ \mathbf{0}&\mathbf{0}&\mathbf{L}&\mathbf{1}\\ \mathbf{1}&\mathbf{0}&\mathbf{0}&\mathbf{0}\end{pmatrix}\begin{pmatrix}\mathbf{L}&\mathbf{1}&\mathbf{0}&\mathbf{0}\\ \mathbf{0}&\mathbf{0}&\mathbf{1}&\mathbf{0}\\ \mathbf{0}&\mathbf{0}&\mathbf{L}&\mathbf{1}\\ \mathbf{1}&\mathbf{0}&\mathbf{0}&\mathbf{0}\end{pmatrix}\begin{pmatrix}\mathbf{1}&\mathbf{1}&\mathbf{0}&\mathbf{0}\\ \mathbf{0}&\mathbf{0}&\mathbf{1}&\mathbf{0}\\ \mathbf{0}&\mathbf{0}&\mathbf{1}&\mathbf{1}\\ \mathbf{1}&\mathbf{0}&\mathbf{0}&\mathbf{0}\end{pmatrix}\\
&= \begin{pmatrix}\mathbf{L}^2+\mathbf{1}&\mathbf{L}^2&\mathbf{1}&\mathbf{L}+\mathbf{1}\\ \mathbf{L}+\mathbf{1}&\mathbf{1}&\mathbf{L}^2&\mathbf{L}^2\\ \mathbf{1}&\mathbf{L}+\mathbf{1}&\mathbf{L}^2+\mathbf{1}&\mathbf{L}^2\\ \mathbf{L}^2&\mathbf{L}^2&\mathbf{L}+\mathbf{1}&\mathbf{1}\end{pmatrix}.
\end{aligned} \tag{10}$$

Then the implementation cost of the matrix $\mathbf{B}$ for $m$-bit input is equal to

$$\overbrace{(2m)}^{\#\mathbf{A}_1}+\overbrace{(2m+2(\#\mathbf{L}))}^{\#\mathbf{A}_2}+\overbrace{(2m+2(\#\mathbf{L}))}^{\#\mathbf{A}_3}+\overbrace{(2m)}^{\#\mathbf{A}_4}=8m+4(\#\mathbf{L}). \tag{11}$$

There are $\binom{8}{4}-1=69$ subdeterminants in $\mathbf{B}$. The base set of these subdeterminants is:

$$\{\mathbf{L},\mathbf{L}+\mathbf{1},\mathbf{L}^2+\mathbf{L}+\mathbf{1},\mathbf{L}^3+\mathbf{L}+\mathbf{1},\mathbf{L}^3+\mathbf{L}^2+\mathbf{1},\mathbf{L}^4+\mathbf{L}+\mathbf{1}\} \tag{12}$$

For $m=4$, consider the following $4\times 4$ non-singular matrices over $\mathbb{F}_2$. It can be verified that by applying $\mathbf{L}_i$ with $1\leq i\leq 4$ , the given elements in (12) are non-singular matrices over $\mathbb{F}_2$. Moreover, the implementation cost of $\mathbf{L}_i$'s is one XOR.

$$\mathbf{L}_1=[[1,4],[1],[2],[3]],\quad \mathbf{L}_2=[[4],[1,2],[2],[3]],\quad \mathbf{L}_3=[[4],[1],[2,3],[3]],\quad \mathbf{L}_4=[[4],[1],[2],[3,4]].$$

Hence, by applying $\mathbf{L}_i$'s and relation (11), $\mathbf{B}$ is implemented with $8\times 4+4\times 1=36$ XOR for 4-bit input. Moreover, $\mathbf{L}^4+\mathbf{L}^3+1$ is not an element of (12). Therefore, $\mathbf{B}$ can be implemented with 36 XOR over $\mathbb{F}_{2^4}/\texttt{0x19}$. Actually, $\mathbf{L}_4$ is the binary matrix representation of the root of `0x19`. Furthermore, by using $\mathbf{L}_i$ with $1\leq i\leq 4$, the implementation cost of $\mathbf{B}^{-1}$ is 36 XOR, since we have

$$\mathbf{B}^{-1}=\begin{pmatrix}\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{1}\\ \mathbf{1}&\mathbf{0}&\mathbf{0}&\mathbf{1}\\ \mathbf{0}&\mathbf{1}&\mathbf{0}&\mathbf{0}\\ \mathbf{0}&\mathbf{1}&\mathbf{1}&\mathbf{0}\end{pmatrix}\begin{pmatrix}\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{1}\\ \mathbf{1}&\mathbf{0}&\mathbf{0}&\mathbf{L}\\ \mathbf{0}&\mathbf{1}&\mathbf{0}&\mathbf{0}\\ \mathbf{0}&\mathbf{L}&\mathbf{1}&\mathbf{0}\end{pmatrix}\begin{pmatrix}\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{1}\\ \mathbf{1}&\mathbf{0}&\mathbf{0}&\mathbf{L}\\ \mathbf{0}&\mathbf{1}&\mathbf{0}&\mathbf{0}\\ \mathbf{0}&\mathbf{L}&\mathbf{1}&\mathbf{0}\end{pmatrix}\begin{pmatrix}\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{1}\\ \mathbf{1}&\mathbf{0}&\mathbf{0}&\mathbf{1}\\ \mathbf{0}&\mathbf{1}&\mathbf{0}&\mathbf{0}\\ \mathbf{0}&\mathbf{1}&\mathbf{1}&\mathbf{0}\end{pmatrix} \tag{13}$$

For $m=8$, consider $8\times 8$ non-singular matrices $\mathbf{L}_j$ with $1\leq j\leq 7$ in (14). It can be verified that by using $\mathbf{L}_j$'s the given elements in (12) are non-singular matrices over $\mathbb{F}_2$.

$$\begin{array}{ll}\mathbf{L}_1=[[2,8],[1],[2],[3],[4],[5],[6],[7]], & \mathbf{L}_2=[[8],[1,3],[2],[3],[4],[5],[6],[7]],\\ \mathbf{L}_3=[[8],[1],[2,4],[3],[4],[5],[6],[7]], & \mathbf{L}_4=[[8],[1],[2],[3,5],[4],[5],[6],[7]],\\ \mathbf{L}_5=[[8],[1],[2],[3],[4,6],[5],[6],[7]], & \mathbf{L}_6=[[8],[1],[2],[3],[4],[5,7],[6],[7]]\\ \mathbf{L}_7=[[8],[1],[2],[3],[4],[5],[6,8],[7]]. & \end{array} \tag{14}$$

The implementation cost of $\mathbf{L}_j$'s is one XOR. Therefore, by using $\mathbf{L}_j$'s and relation (11), $\mathbf{B}$ is implemented with $8\times 8+4\times 1=68$ XOR for 8-bit input. Moreover, it follows from (13) that the implementation cost of $\mathbf{B}^{-1}$ is 68 XOR for 8-bit input. Furthermore, $\mathbf{B}$ can be implemented with at least $8\times 8+4\times 2=72$ XOR over $\mathbb{F}_{2^8}$, since some roots of irreducible polynomials of degree 8 require only two XOR [Bei16].

Now consider the $2^k\times 2^k$ companion binary matrix $\mathbf{L}_{(k)}=[[2^{k-2},2^k],[1],[2],\cdots,[2^k-1]]$ with $k>1$. It is easy to show that the characteristic polynomial of $\mathbf{L}_{(k)}$ over $\mathbb{F}_2$ is $\left(x^4+x^3+1\right)^{2^{k-2}}$. Moreover, by applying $\mathbf{L}_{(k)}$ the given elements in (12) are non-singular matrices over $\mathbb{F}_2$. Furthermore, the implementation cost of $\mathbf{L}_{(k)}$ is one XOR. Hence, using $\mathbf{L}_{(k)}$ and relation (11) the implementation cost of $\mathbf{B}$ is $8\times 2^k+4$ XOR for $2^k$-bit input with $k>1$. For instance, the implementation cost of $\mathbf{B}$ and $\mathbf{B}^{-1}$ by using $\mathbf{L}_{(6)}$ is $8\times 64+4=516$ XOR for 64-bit input. Notice that if we wanted to implement $\mathbf{B}$ over finite field for 64-bit input, we need to work with the finite field $\mathbb{F}_{2^{64}}$.

## 5.2 Lightweight $8\times 8$ MDS Matrices from Primitive GFS Matrices

First of all, we select two permutations $\widehat{\mathbf{p}}_1=\{1,3,2,4\}$ and $\widehat{\mathbf{p}}_2=\{2,1,4,3\}$. Then we construct the primitive GFS matrix $\mathbf{C}=\mathbf{R}(4,\mathbf{1}_{(m)},[\widehat{\mathbf{p}}_1,\widehat{\mathbf{p}}_2])$. It follows from relation (6) that $\mathbf{C}$ is an 6-primitive GFS matrix over $\mathbb{R}$. In fact, based on the Appendix A, the GFS matrix $\mathbf{C}$ has minimum primitive order. Moreover, it follows from complete search that there are no primitive GFS matrices $\mathbf{A}_i=\mathbf{S}(4,\mathbf{f}^{(i)}_{(m)},[\widehat{\mathbf{p}}_1,\widehat{\mathbf{p}}_2])$ with $1\leq i\leq 6$ such that

$\prod_{i=1}^{6} \mathbf{A}_i$ be an MDS matrix over $\mathbb{F}_2[\mathbf{L}]$ provided that $\mathbf{f}^{(i)} \in \{\mathbf{L}^{-2}, \mathbf{L}^{-1}, \mathbf{1}, \mathbf{L}, \mathbf{L}^{-2}\}$. Now consider the following primitive GFS matrices $\mathbf{A}_i$ with $1 \leq i \leq 7$.

$$\begin{array}{llll} \mathbf{A}_1 = \mathbf{R}(4, \mathbf{L}_{(m)}, [\widehat{\mathbf{p}}_1, \widehat{\mathbf{p}}_2]), & \mathbf{A}_2 = \mathbf{R}(4, \mathbf{1}_m, [\widehat{\mathbf{p}}_1, \widehat{\mathbf{p}}_2]), & \mathbf{A}_3 = \mathbf{R}(4, \mathbf{1}_m, [\widehat{\mathbf{p}}_1, \widehat{\mathbf{p}}_2]), & \mathbf{A}_4 = \mathbf{R}(4, \mathbf{L}^2_{(m)}, [\widehat{\mathbf{p}}_1, \widehat{\mathbf{p}}_2]), \\ \mathbf{A}_5 = \mathbf{R}(4, \mathbf{L}^{-1}_{(m)}, [\widehat{\mathbf{p}}_1, \widehat{\mathbf{p}}_2]), & \mathbf{A}_6 = \mathbf{R}(4, \mathbf{L}^{-1}_{(m)}, [\widehat{\mathbf{p}}_1, \widehat{\mathbf{p}}_2]), & \mathbf{A}_7 = \mathbf{R}(4, \mathbf{1}_m, [\widehat{\mathbf{p}}_1, \widehat{\mathbf{p}}_2]). & \end{array}$$

It can be verified that $\mathbf{B} = \prod_{i=1}^{7} \mathbf{A}_i$ is an MDS matrix over $\mathbb{F}_2[\mathbf{L}]$. Moreover, using a complete search, $\mathbf{B}$ is one of the optimal results in relation to the number of binary linear functions ($\mathbf{L}$). Actually, it is not possible to construct an MDS matrix $\mathbf{B}$ by applying primitive GFS matrices $\mathbf{A}_i = \mathbf{S}(4, \mathbf{f}^{(i)}_{(m)}, [\widehat{\mathbf{p}}_1, \widehat{\mathbf{p}}_2])$ with $1 \leq i \leq 7$ provided that $\mathbf{A}_i$'s are constructed from less than twenty $\mathbf{L}$. The matrix structure $\mathbf{B}$ is as follows

$$\begin{aligned} \mathbf{B} &= \mathbf{A}_1\, \mathbf{A}_2\, \mathbf{A}_3\, \mathbf{A}_4\, \mathbf{A}_5\, \mathbf{A}_6\, \mathbf{A}_7 \\ &= \begin{pmatrix} \mathbf{L}^3+\mathbf{L}^{-1}+\mathbf{1} & \mathbf{L}^3+\mathbf{L}+\mathbf{L}^{-1} & \mathbf{L}^2+\mathbf{L}+\mathbf{1} & \mathbf{L} \\ \mathbf{L}^2+\mathbf{L}^{-2}+\mathbf{1} & \mathbf{L}^2+\mathbf{L}^{-2} & \mathbf{L}^2+\mathbf{L}^{-1}+\mathbf{1} & \mathbf{L}^2+\mathbf{1} \\ \mathbf{L}^2+\mathbf{L}+\mathbf{1} & \mathbf{L} & \mathbf{L}^{-2}+\mathbf{1} & \mathbf{L}^2+\mathbf{L}+\mathbf{L}^{-2}+\mathbf{1} \\ \mathbf{L}^2+\mathbf{L}^{-1}+\mathbf{1} & \mathbf{L}^2+\mathbf{1} & \mathbf{L}+\mathbf{L}^{-1}+\mathbf{1} & \mathbf{L}^{-1}+\mathbf{1} \\ \mathbf{L}^3+\mathbf{L}+\mathbf{1} & \mathbf{L}^3+\mathbf{L}^{-1}+\mathbf{1} & \mathbf{L}^3+\mathbf{L}^{-1}+\mathbf{1} & \mathbf{L}^3+\mathbf{L}+\mathbf{L}^{-1} \\ \mathbf{L}+\mathbf{L}^{-1} & \mathbf{L}^{-1}+\mathbf{1} & \mathbf{L}^2+\mathbf{L}^{-2}+\mathbf{1} & \mathbf{L}^2+\mathbf{L}^{-2} \\ \mathbf{L}^{-2}+\mathbf{1} & \mathbf{L}^2+\mathbf{L}+\mathbf{L}^{-2}+\mathbf{1} & \mathbf{L}^3+\mathbf{L}+\mathbf{1} & \mathbf{L}^3+\mathbf{L}^{-1}+\mathbf{1} \\ \mathbf{L}+\mathbf{L}^{-1}+\mathbf{1} & \mathbf{L}^{-1}+\mathbf{1} & \mathbf{L}+\mathbf{L}^{-1} & \mathbf{L}^{-1}+\mathbf{1} \end{pmatrix} \\ &\quad \begin{pmatrix} \mathbf{L}^3+\mathbf{L}+\mathbf{1} & \mathbf{L}^3+\mathbf{L}^{-1}+\mathbf{1} & \mathbf{L}^{-2}+\mathbf{1} & \mathbf{L}^2+\mathbf{L}+\mathbf{L}^{-2}+\mathbf{1} \\ \mathbf{L}+\mathbf{L}^{-1} & \mathbf{L}^{-1}+\mathbf{1} & \mathbf{L}+\mathbf{L}^{-1}+\mathbf{1} & \mathbf{L}^{-1}+\mathbf{1} \\ \mathbf{L}^3+\mathbf{L}^{-1}+\mathbf{1} & \mathbf{L}^3+\mathbf{L}+\mathbf{L}^{-1} & \mathbf{L}^3+\mathbf{L}+\mathbf{1} & \mathbf{L}^3+\mathbf{L}^{-1}+\mathbf{1} \\ \mathbf{L}^2+\mathbf{L}^{-2}+\mathbf{1} & \mathbf{L}^2+\mathbf{L}^{-2} & \mathbf{L}+\mathbf{L}^{-1} & \mathbf{L}^{-1}+\mathbf{1} \\ \mathbf{L}^{-2}+\mathbf{1} & \mathbf{L}^2+\mathbf{L}+\mathbf{L}^{-2}+\mathbf{1} & \mathbf{L}^2+\mathbf{L}+\mathbf{1} & \mathbf{L} \\ \mathbf{L}+\mathbf{L}^{-1}+\mathbf{1} & \mathbf{L}^{-1}+\mathbf{1} & \mathbf{L}^2+\mathbf{L}^{-1}+\mathbf{1} & \mathbf{L}^2+\mathbf{1} \\ \mathbf{L}^2+\mathbf{L}+\mathbf{1} & \mathbf{L} & \mathbf{L}^3+\mathbf{L}^{-1}+\mathbf{1} & \mathbf{L}^3+\mathbf{L}+\mathbf{L}^{-1} \\ \mathbf{L}^2+\mathbf{L}^{-1}+\mathbf{1} & \mathbf{L}^2+\mathbf{1} & \mathbf{L}^2+\mathbf{L}^{-2}+\mathbf{1} & \mathbf{L}^2+\mathbf{L}^{-2} \end{pmatrix}. \end{aligned} \tag{15}$$

It follows from (15) that the implementation cost of $\mathbf{B}$ for $m$-bit input is equal to

$$\begin{aligned} &\overbrace{(4m+4(\#\mathbf{L}))}^{\#\mathbf{A}_1} + \overbrace{(4m)}^{\#\mathbf{A}_2} + \overbrace{(4m)}^{\#\mathbf{A}_3} + \overbrace{(4m+4(\#\mathbf{L}^2))}^{\#\mathbf{A}_4} + \overbrace{(4m+4(\#\mathbf{L}^{-1}))}^{\#\mathbf{A}_5} + \overbrace{(4m+4(\#\mathbf{L}^{-1}))}^{\#\mathbf{A}_6} + \overbrace{(4m)}^{\#\mathbf{A}_7} \\ &= 28m + 4(\#\mathbf{L}) + 4(\#\mathbf{L}^2) + 8(\#\mathbf{L}^{-1}) \end{aligned} \tag{16}$$

The base set of subdeterminants of $\mathbf{B}$ has 380 elements. In this base set, there are all irreducible polynomials of degree 4 and 8, except for the irreducible polynomial `0x1A3`. It can be checked that the elements of this base set are non-singular matrices over $\mathbb{F}_2$ by applying the following non-singular $8 \times 8$ binary matrices $\mathbf{L}_i$ with $1 \leq i \leq 3$.

$$\begin{aligned} &\mathbf{L}_1 = [[1,3,7,8],[1],[2],[3],[4],[5],[6],[7]], \quad \mathbf{L}_2 = [[8],[1,2,4,8],[2],[3],[4],[5],[6],[7]], \\ &\mathbf{L}_3 = [[8],[1,8],[2],[3],[4],[5,8],[6],[7,8]] \end{aligned} \tag{17}$$

First notice that the implementation cost of $\mathbf{L}_i$ and $\mathbf{L}_i^{-1}$ with $1 \leq i \leq 3$ is three XOR. In addition, the implementation cost of $\mathbf{L}_i^2$ with $1 \leq i \leq 3$ is seven XOR. But $\mathbf{L}_i^2$'s can be implemented with five XOR. For instance, consider $\mathbf{x} = [x_1, x_2, \cdots, x_8]$. Then we have

$$\begin{aligned} &\mathbf{L}_3^2 \cdot \mathbf{x}^T = [x_7 + x_8, x_7, x_1 + x_8, x_2, x_3, x_4 + x_7 + x_8, x_5 + x_8, x_6 + x_7 + x_8]^T, \\ &u_1 = x_7 + x_8,\ u_2 = x_1 + x_8,\ u_3 = u_1 + x_4, u_4 = x_5 + x_8,\ u_5 = u_1 + x_6. \end{aligned}$$

Therefore, by applying $\mathbf{L}_i$'s and relation (16), $\mathbf{B}$ is implemented with $28 \times 8 + 4 \times 3 + 4 \times 5 + 8 \times 3 = 280$ XOR for 8-bit input. Moreover, it can be verify that the implementation cost of $\mathbf{B}^{-1}$ is 280 XOR for 8-bit input. Furthermore, $\mathbf{B}$ is implemented with 280 XOR over $\mathbb{F}_{2^8}/$`0x1A3`, since $\mathbf{L}_3$ is the binary matrix representation of the root of `0x1A3`.

Consider $\mathbf{L}$ is an $2^k \times 2^k$ with $k > 3$ companion binary matrix provided that the characteristic polynomial of $\mathbf{L}$ over $\mathbb{F}_2$ is $\left(x^8 + x^7 + x^5 + x + 1\right)^{2^{k-3}}$. It is not difficult to prove

that by applying $\mathbf{L}$ the elements of the base set of subdeterminants of $\mathbf{B}$ in (15), are non-singular matrices over $\mathbb{F}_2$. Moreover, the implementation cost of $\mathbf{L}$ and $\mathbf{L}^{-1}$ is three XOR and also $\mathbf{L}^2$ can be implemented with six XOR. Hence, using $\mathbf{L}$ and relation (16) the implementation cost of $\mathbf{B}$ and $\mathbf{B}^{-1}$ are $28 \times 2^k + 60$ XOR for $2^k$-bit input with $k > 3$. For instance, $\mathbf{L} = [[32], [1], [2], [3], [4, 32], [5], \cdots, [19], [20, 32], [21], \cdots, [27], [28, 32], [29], [30], [31]]$ is an $32 \times 32$ companion binary matrix such that its characteristic polynomial is $\left(x^8 + x^7 + x^5 + x + 1\right)^4$ over $\mathbb{F}_2$. Therefore, the implementation cost of $\mathbf{B}$ by applying $\mathbf{L}$ is $28 \times 32 + 60 = 956$ XOR for 32-bit input.

# 6 Construction of Lightweight MDS Matrices by Applying Extended Primitive GFS Matrices

This section is the main work of this paper. First of all, by extension of Definition 2 we define some type of sparse matrices called EGFS matrices. Then, by applying EGFS matrices we propose $4 \times 4$, $6 \times 6$ and $8 \times 8$ lightweight MDS matrices with implementation cost 67, 158 and 272 XOR for 8-bit input, respectively. The proposed MDS matrices are not only suitable by terminology of implementation cost, but also are efficient with respect to the number of binary linear functions that are used in construction of these matrices.

The proposed $4 \times 4$ MDS matrix is obtained by complete search. But a random search is applied to achieve the proposed $6 \times 6$ and $8 \times 8$ MDS matrices. Actually, there is an extensive search space to obtain lightweight $6 \times 6$ and $8 \times 8$ MDS matrices using EGFS matrices. Therefore, by performing a full search on EGFS matrices, better implementation results may be achieved to construct $6 \times 6$ and $8 \times 8$ lightweight MDS matrices.

Assume that $\mathbf{1}$ and $\mathbf{0}$ are $m \times m$ identity and zero matrices over $\mathbb{F}_2$, receptively. Consider $\mathbf{L}_j \in \{\mathbf{1}, \mathbf{L}, \mathbf{L}^{-1}\}$ with $1 \leq j \leq 3n$ such that $\mathbf{L}$ is an $m \times m$ non-singular matrix over $\mathbb{F}_2$. In Definition 3, the following $2 \times 2$ block-matrices with $1 \leq i \leq n$ are used.

$$\mathbf{c}_i^{(1,m)} = \begin{pmatrix} \mathbf{L}_{3i-2} & \mathbf{L}_{3i-1} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}, \quad \mathbf{c}_i^{(2,m)} = \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{L}_{3i} & \mathbf{0} \end{pmatrix}, \quad \mathbf{z}^{(m)} = \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}. \tag{18}$$

**Definition 3** (Extended Primitive GFS Matrices)**.** Consider $\mathbf{p}_1 = \{a_1, a_2, \cdots, a_n\}$ and $\mathbf{p}_2 = \{b_1, b_2, \cdots, b_n\}$ are two permutations from 1 to $n$ provided that $a_i \neq b_i$ for $1 \leq i \leq n$. Consider $2 \times 2$ block-matrices $\mathbf{c}_i^{(1,m)}$, $\mathbf{c}_i^{(2,m)}$ and $\mathbf{z}^{(m)}$ with $1 \leq i \leq n$ that are given in (18).

Assume that by using two permutations $\mathbf{p}_1$ and $\mathbf{p}_2$, the $i$th row of an $n \times n$ block-matrix $\mathbf{E} = (e_{i,j})$ with $1 \leq i, j \leq n$, is filled in the following form.

$$e_{i,j} = \begin{cases} \mathbf{c}_i^{(1,m)} & j = a_i, \\ \mathbf{c}_i^{(2,m)} & j = b_i, \\ \mathbf{z}^{(m)} & j \notin \{a_i, b_i\}. \end{cases}$$

The block-matrix $\mathbf{E}$ is called an extended primitive GFS matrix, denoted with EGFS matrix, if $\mathbf{E}$ be a primitive matrix over $\mathbb{R}$.

In Definition 3 to check whether $\mathbf{E}$ is a primitive matrix over $\mathbb{R}$, we suppose that $\mathbf{c}_i^{(1,m)}$, $\mathbf{c}_i^{(2,m)}$ and $\mathbf{z}^{(m)}$ with $1 \leq i \leq n$ are $2 \times 2$ matrices over $\mathbb{R}$ and also $\mathbf{L}_j$ with $1 \leq j \leq 3n$ are positive integer numbers over $\mathbb{R}$.

Moreover, it can be verified that $\det(\mathbf{E}) = \prod_{i=1}^{n} \mathbf{L}_{3i-1}\mathbf{L}_{3i}$ over $\mathbb{F}_2[\mathbf{L}]$. Therefore, $\det(\mathbf{E}) \neq 0$ over $\mathbb{F}_2[\mathbf{L}]$, since it is assumed that $\mathbf{L}_j \in \{\mathbf{1}, \mathbf{L}, \mathbf{L}^{-1}\}$ with $1 \leq j \leq 3n$. In other words, EGFS matrices are non-singular matrices over $\mathbb{F}_2[\mathbf{L}]$.

**Example 6.** For $n = 2$, consider two permutations $\mathbf{p}_1 = \{1, 2\}$ and $\mathbf{p}_2 = \{2, 1\}$. Then the following two EGFS matrices $\mathbf{E}_1$ and $\mathbf{E}_2$ are 4-primitive matrices over $\mathbb{R}$.

$$\mathbf{c}_1^{(1,m)} = \begin{pmatrix} \mathbf{1} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}, \mathbf{c}_2^{(1,m)} = \begin{pmatrix} \mathbf{L} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}, \qquad \mathbf{c}_1^{(1,m)} = \begin{pmatrix} \mathbf{L} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}, \mathbf{c}_2^{(1,m)} = \begin{pmatrix} \mathbf{1} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} \end{pmatrix},$$

$$\mathbf{c}_1^{(2,m)} = \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{1} & \mathbf{0} \end{pmatrix}, \mathbf{c}_2^{(2,m)} = \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{L}^{-1} & \mathbf{0} \end{pmatrix}, \qquad \mathbf{c}_1^{(2,m)} = \mathbf{c}_2^{(2,m)} = \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{1} & \mathbf{0} \end{pmatrix},$$

$$\mathbf{E}_1 = \left(\begin{array}{cc|cc} \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{L} & \mathbf{1} \\ \mathbf{L}^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{array}\right). \qquad \mathbf{E}_2 = \left(\begin{array}{cc|cc} \mathbf{L} & \mathbf{1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{1} \\ \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{array}\right).$$

For $n = 3$, the given EGFS matrix in (19) is an 6-primitive matrix over $\mathbb{R}$.

$$\begin{aligned} &\mathbf{c}_1^{(1,m)} = \mathbf{c}_2^{(1,m)} = \mathbf{c}_3^{(1,m)} = \begin{pmatrix} \mathbf{L} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}, \\ &\mathbf{c}_1^{(2,m)} = \mathbf{c}_2^{(2,m)} = \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{1} & \mathbf{0} \end{pmatrix}, \mathbf{c}_3^{(2,m)} = \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{L}^{-1} & \mathbf{0} \end{pmatrix}, \\ &\mathbf{p}_1 = \{3, 1, 2\}, \quad \mathbf{p}_2 = \{2, 3, 1\}. \end{aligned} \qquad \mathbf{E} = \left(\begin{array}{cc|cc|cc} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{L} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{L} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{L} & \mathbf{1} & \mathbf{0} & \mathbf{0} \\ \mathbf{L}^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{array}\right). \tag{19}$$

## 6.1 Construction of Lightweight $4 \times 4$ MDS Matrices from EGFS

In this subsection, we propose four EGFS matrices $\mathbf{E}_i$ with $1 \leq i \leq 4$ such that $\mathbf{E}_i$'s are 4-primitive matrices over $\mathbb{R}$. Moreover, proposed EGFS matrices are constructed of two permutations $\mathbf{p}_1 = \{1, 2\}$ and $\mathbf{p}_2 = \{2, 1\}$. In other words, $\mathbf{E}_i$'s are with the same structure. In the rest, by applying $\mathbf{E}_i$'s we obtain an MDS matrix $\mathbf{H}$ over $\mathbb{F}_2[\mathbf{L}]$ such that the implementation cost of $\mathbf{H}$ over 4 and 8-bit input are 35 and 67 XOR, respectively.

$$\begin{aligned} \mathbf{H} &= \mathbf{E}_1\, \mathbf{E}_2\, \mathbf{E}_3\, \mathbf{E}_4 \\ &= \begin{pmatrix} \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{1} \\ \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{L} & \mathbf{1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{1} \\ \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{L} & \mathbf{1} \\ \mathbf{L}^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{1} \\ \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{pmatrix} \\ &= \begin{pmatrix} \mathbf{L}+\mathbf{1} & \mathbf{L} & \mathbf{1} & \mathbf{L}+\mathbf{1} \\ \mathbf{L}^{-1}+\mathbf{1} & \mathbf{L}^{-1} & \mathbf{L} & \mathbf{L} \\ \mathbf{L}^{-1} & \mathbf{L}^{-1}+\mathbf{1} & \mathbf{L}+\mathbf{1} & \mathbf{L} \\ \mathbf{L} & \mathbf{L} & \mathbf{L}+\mathbf{1} & \mathbf{1} \end{pmatrix}. \end{aligned}$$

The implementation cost of $\mathbf{H}$ for $m$-bit input is equal to

$$\overbrace{(2m)}^{\#\mathbf{E}_1} + \overbrace{(2m + \#\mathbf{L})}^{\#\mathbf{E}_2} + \overbrace{(2m + \#\mathbf{L} + \#(\mathbf{L}^{-1}))}^{\#\mathbf{E}_3} + \overbrace{(2m)}^{\#\mathbf{E}_4} = 8m + 2(\#\mathbf{L}) + \#(\mathbf{L}^{-1}). \tag{20}$$

The base set of subdeterminants of $\mathbf{H}$ is:

$$\{\mathbf{L}, \mathbf{L}+\mathbf{1}, \mathbf{L}^2+\mathbf{L}+\mathbf{1}, \mathbf{L}^3+\mathbf{L}+\mathbf{1}, \mathbf{L}^3+\mathbf{L}^2+\mathbf{1}\}. \tag{21}$$

For $m = 4$, consider $4 \times 4$ non-singular binary matrices $\mathbf{L}_i$ with $1 \leq i \leq 6$ in (22). It can be checked that by using $\mathbf{L}_i$'s the given elements in (21) are non-singular matrices over $\mathbb{F}_2$. In addition, the implementation cost of $\mathbf{L}_i$ and $\mathbf{L}_i^{-1}$ with $1 \leq i \leq 6$ are one XOR.

$$\begin{aligned} &\mathbf{L}_1 = [[1, 4], [1], [2], [3]], \quad \mathbf{L}_2 = [[3, 4], [1], [2], [3]], \quad \mathbf{L}_3 = [[4], [1, 2], [2], [3]], \\ &\mathbf{L}_4 = [[4], [1, 4], [2], [3]], \quad \mathbf{L}_5 = [[4], [1], [2, 3], [3]], \quad \mathbf{L}_6 = [[4], [1], [2], [3, 4]]. \end{aligned} \tag{22}$$

Therefore, by applying $\mathbf{L}_i$'s and relation (20), the implementation cost of $\mathbf{H}$ is $8\times 4+2\times 1+1\times 1=35$ XOR for 4-bit input. Moreover, it follows from (21) that the matrix $\mathbf{H}$ can be implemented with 35 XOR over $\mathbb{F}_{2^4}$, since two irreducible polynomials `0x13` and `0x19` are not elements of the set (21).

For $m=8$, the next $8\times 8$ non-singular binary matrices $\mathbf{L}_j$ with $1\leq j\leq 10$ are obtained such that the given elements in (21) are non-singular matrices over $\mathbb{F}_2$ by applying $\mathbf{L}_j$'s. Moreover, the implementation cost of $\mathbf{L}_j$ and $\mathbf{L}_j^{-1}$ with $1\leq j\leq 10$ are one XOR.

$$\begin{array}{ll}\mathbf{L}_1=[[2,8],[1],[2],[3],[4],[5],[6],[7]], & \mathbf{L}_2=[[6,8],[1],[2],[3],[4],[5],[6],[7]],\\ \mathbf{L}_3=[[8],[1,3],[2],[3],[4],[5],[6],[7]], & \mathbf{L}_4=[[8],[1,7],[2],[3],[4],[5],[6],[7]],\\ \mathbf{L}_5=[[8],[1],[2,4],[3],[4],[5],[6],[7]], & \mathbf{L}_6=[[8],[1],[2,8],[3],[4],[5],[6],[7]],\\ \mathbf{L}_7=[[8],[1],[2],[3,5],[4],[5],[6],[7]], & \mathbf{L}_8=[[8],[1],[2],[3],[4,6],[5],[6],[7]],\\ \mathbf{L}_9=[[8],[1],[2],[3],[4],[5,7],[6],[7]], & \mathbf{L}_{10}=[[8],[1],[2],[3],[4],[5],[6,8],[7]].\end{array} \tag{23}$$

Hence, it follows from (20) that the implementation cost of $\mathbf{H}$ by applying $\mathbf{L}_j$'s is $8\times 8+2\times 1+1\times 1=67$ XOR. Moreover, some roots of irreducible polynomials of degree 8 require only two XOR [Bei16]. Therefore, $\mathbf{H}$ can be implemented with at least $8\times 8+2\times 2+1\times 2=70$ XOR over $\mathbb{F}_{2^8}$, since there are no irreducible polynomials in $\mathbb{F}_2$ of degree 8 in (21).

## 6.2 Construction of Lightweight $6\times 6$ MDS Matrices from EGFS

In this subsection, by applying EGFS matrices, we propose a new lightweight $6\times 6$ MDS matrix which is implemented with 158 XOR for 8-bit input. The result of this subsection is not only efficient from hardware perspective, but also are suitable by software point of view. Actually, in order to construct the proposed $6\times 6$ MDS matrix, seven binary linear functions ($\mathbf{L}$) are used. Moreover, the cost of $\mathbf{L}$ is very small compared to the total XOR.

Consider the following EGFS matrices $\mathbf{E}_i$ with $1\leq i\leq 3$ such that $\mathbf{E}_i$'s are constructed from two permutations $\mathbf{p}_1=\{3,1,2\}$ and $\mathbf{p}_2=\{2,3,1\}$. It follows from (19) that $\mathbf{E}_i$'s are 6-primitive matrices over $\mathbb{R}$, since $\mathbf{E}_i$'s are with the same structure as $\mathbf{E}$ in (19).

$$\mathbf{E}_1=\begin{pmatrix}\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{1}&\mathbf{1}\\ \mathbf{0}&\mathbf{0}&\mathbf{1}&\mathbf{0}&\mathbf{0}&\mathbf{0}\\ \mathbf{1}&\mathbf{1}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}\\ \mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{1}&\mathbf{0}\\ \mathbf{0}&\mathbf{0}&\mathbf{1}&\mathbf{1}&\mathbf{0}&\mathbf{0}\\ \mathbf{1}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}\end{pmatrix},\mathbf{E}_2=\begin{pmatrix}\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{L}&\mathbf{1}\\ \mathbf{0}&\mathbf{0}&\mathbf{1}&\mathbf{0}&\mathbf{0}&\mathbf{0}\\ \mathbf{L}&\mathbf{1}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}\\ \mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{1}&\mathbf{0}\\ \mathbf{0}&\mathbf{0}&\mathbf{L}&\mathbf{1}&\mathbf{0}&\mathbf{0}\\ \mathbf{1}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}\end{pmatrix},\mathbf{E}_3=\begin{pmatrix}\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{L}&\mathbf{1}\\ \mathbf{0}&\mathbf{0}&\mathbf{1}&\mathbf{0}&\mathbf{0}&\mathbf{0}\\ \mathbf{L}&\mathbf{1}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}\\ \mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{1}&\mathbf{0}\\ \mathbf{0}&\mathbf{0}&\mathbf{L}&\mathbf{1}&\mathbf{0}&\mathbf{0}\\ \mathbf{1/L}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}\end{pmatrix}.$$

Now based on the EGFS matrices $\mathbf{E}_i$ with $1\leq i\leq 3$, the proposed lightweight $6\times 6$ MDS matrix $\mathbf{H}$ is constructed as follows.

$$\mathbf{H}=\mathbf{E}_1^2\times\mathbf{E}_2\times\mathbf{E}_3\times\mathbf{E}_1^2=\begin{pmatrix}\mathbf{L^2+L^{-1}}&\mathbf{L^2}&\mathbf{L^2+L^{-1}}&\mathbf{L+L^{-1}+1}&\mathbf{1}&\mathbf{L^2+1}\\ \mathbf{L+L^{-1}+1}&\mathbf{1}&\mathbf{L^2+L^{-1}}&\mathbf{L^{-1}}&\mathbf{L^2}&\mathbf{L^2}\\ \mathbf{1}&\mathbf{L^2+1}&\mathbf{L^2+1}&\mathbf{L^2}&\mathbf{L^2+1}&\mathbf{L}\\ \mathbf{L^2}&\mathbf{L^2}&\mathbf{L}&\mathbf{1}&\mathbf{L^2+1}&\mathbf{1}\\ \mathbf{L^2+L^{-1}}&\mathbf{L}&\mathbf{L^{-1}}&\mathbf{L^2+L^{-1}}&\mathbf{L^2+1}&\mathbf{L^2}\\ \mathbf{L^2+1}&\mathbf{1}&\mathbf{L^2}&\mathbf{L^2}&\mathbf{L}&\mathbf{1}\end{pmatrix}.$$

The implementation cost of $\mathbf{H}$ for $m$-bit input is equal to

$$\overbrace{2(3m)}^{2(\#\mathbf{E}_1)}+\overbrace{(3m+3(\#\mathbf{L}))}^{\#\mathbf{E}_2}+\overbrace{(3m+3(\#\mathbf{L})+\#(\mathbf{L}^{-1}))}^{\#\mathbf{E}_3}+\overbrace{2(3m)}^{2(\#\mathbf{E}_1)}=18m+6(\#\mathbf{L})+\#(\mathbf{L}^{-1}) \tag{24}$$

The base set of subdeterminants of $\mathbf{H}$ has 35 elements that are listed in (25).

$$\begin{gathered}\{L, L+1, L^2+L+1, L^3+L+1, L^3+L^2+1, L^4+L+1, L^4+L^3+1,\\ L^4+L^3+L^2+L+1, L^5+L^2+1, L^5+L^3+1, L^5+L^3+L^2+L+1,\\ L^5+L^4+L^2+L+1, L^5+L^4+L^3+L+1, L^5+L^4+L^3+L^2+1,\\ L^6+L+1, L^6+L^3+1, L^6+L^4+L^2+L+1, L^6+L^4+L^3+L+1,\\ L^6+L^5+1, L^6+L^5+L^2+L+1, L^6+L^5+L^3+L^2+1,\\ L^6+L^5+L^4+L+1, L^6+L^5+L^4+L^2+1, L^7+L+1,\\ L^7+L^3+1, L^7+L^3+L^2+L+1, L^7+L^4+1, L^7+L^5+L^2+L+1,\\ L^7+L^5+L^4+L^3+1, L^7+L^5+L^4+L^3+L^2+L+1,\\ L^7+L^6+1, L^7+L^6+L^3+L+1, L^7+L^6+L^4+L+1,\\ L^7+L^6+L^4+L^2+1, L^7+L^6+L^5+L^4+L^3+L^2+1\}\end{gathered} \tag{25}$$

Consider $8\times 8$ non-singular binary matrices $\mathbf{L}_i$ with $1\le i\le 32$ that are given in (26). It can be verified that by applying $\mathbf{L}_i$'s the given elements in (25) are non-singular matrices over $\mathbb{F}_2$. Moreover, the implementation cost of $\mathbf{L}_i$ and $\mathbf{L}_i^{-1}$ with $1\le i\le 32$ are two XOR.

$$\begin{array}{ll}
\mathbf{L}_1=[[1,8],[1],[2,5],[3],[4],[5],[6],[7]], & \mathbf{L}_2=[[1,8],[1],[2,7],[3],[4],[5],[6],[7]],\\
\mathbf{L}_3=[[1,8],[1],[2],[3,6],[4],[5],[6],[7]], & \mathbf{L}_4=[[1,8],[1],[2],[3],[4,7],[5],[6],[7]],\\
\mathbf{L}_5=[[2,8],[1],[2],[3,6],[4],[5],[6],[7]], & \mathbf{L}_6=[[2,8],[1],[2],[3],[4,7],[5],[6],[7]],\\
\mathbf{L}_7=[[3,8],[1],[2],[3],[4,5],[5],[6],[7]], & \mathbf{L}_8=[[3,8],[1],[2],[3],[4,6],[5],[6],[7]],\\
\mathbf{L}_9=[[3,8],[1],[2],[3],[4],[5,6],[6],[7]], & \mathbf{L}_{10}=[[3,8],[1],[2],[3],[4],[5,7],[6],[7]],\\
\mathbf{L}_{11}=[[3,8],[1],[2],[3],[4],[5],[6,7],[7]], & \mathbf{L}_{12}=[[5,8],[1],[2],[3],[4],[5],[6,7],[7]],\\
\mathbf{L}_{13}=[[8],[1,2],[2],[3,6],[4],[5],[6],[7]], & \mathbf{L}_{14}=[[8],[1,2],[2],[3,8],[4],[5],[6],[7]],\\
\mathbf{L}_{15}=[[8],[1,2],[2],[3],[4,7],[5],[6],[7]], & \mathbf{L}_{16}=[[8],[1,2],[2],[3],[4],[5,8],[6],[7]],\\
\mathbf{L}_{17}=[[8],[1,3],[2],[3],[4,7],[5],[6],[7]], & \mathbf{L}_{18}=[[8],[1,3],[2],[3],[4],[5,8],[6],[7]],\\
\mathbf{L}_{19}=[[8],[1,4],[2],[3],[4],[5,6],[6],[7]], & \mathbf{L}_{20}=[[8],[1,4],[2],[3],[4],[5,7],[6],[7]],\\
\mathbf{L}_{21}=[[8],[1,4],[2],[3],[4],[5],[6,7],[7]], & \mathbf{L}_{22}=[[8],[1,4],[2],[3],[4],[5],[6,8],[7]],\\
\mathbf{L}_{23}=[[8],[1,4],[2],[3],[4],[5],[6],[7,8]], & \mathbf{L}_{24}=[[8],[1,6],[2],[3],[4],[5],[6],[7,8]],\\
\mathbf{L}_{25}=[[8],[1],[2,3],[3],[4,7],[5],[6],[7]], & \mathbf{L}_{26}=[[8],[1],[2,3],[3],[4],[5,8],[6],[7]],\\
\mathbf{L}_{27}=[[8],[1],[2,4],[3],[4],[5,8],[6],[7]], & \mathbf{L}_{28}=[[8],[1],[2,5],[3],[4],[5],[6,7],[7]],\\
\mathbf{L}_{29}=[[8],[1],[2,5],[3],[4],[5],[6,8],[7]], & \mathbf{L}_{30}=[[8],[1],[2,5],[3],[4],[5],[6],[7,8]],\\
\mathbf{L}_{31}=[[8],[1],[2],[3,4],[4],[5,8],[6],[7]], & \mathbf{L}_{32}=[[8],[1],[2],[3,6],[4],[5],[6],[7,8]].
\end{array} \tag{26}$$

Therefore, by applying $\mathbf{L}_i$'s and relation (24), $\mathbf{H}$ is implemented with $18\times 8+6\times 2+2=158$ XOR for 8-bit input. Moreover, $\mathbf{H}$ can be implemented with 158 XOR over $\mathbb{F}_{2^8}$, since some roots of irreducible polynomials of degree 8 require only two XOR and there are no irreducible polynomials of degree 8 in (25).

## 6.3 Construction of Lightweight $8\times 8$ MDS Matrices from EGFS

The given results in this subsection is the main result of this paper. Actually, by applying EGFS matrices, we propose an $8\times 8$ MDS matrix $\mathbf{H}$ such that $\mathbf{H}$ is implemented with 272 XOR for 8-bit input. Moreover, the depth of $\mathbf{H}$ is 9 which means $\mathbf{H}$ is an efficient MDS matrix from hardware implementation point of view.

First of all, by applying the Appendix A, we tried to obtain $8\times 8$ EGFS matrices $\mathbf{E}_i$ with $1\le i\le 6$ such that $\mathbf{E}_i$'s satisfy the following conditions. First, $\mathbf{E}_i$'s are with the same structure and are 6-primitive matrices over $\mathbb{R}$. Second, the multiplication of $\mathbf{E}_i$'s denoted with $\mathbf{H}=\prod_{i=1}^{6}\mathbf{E}_i$, is an MDS matrix over $\mathbb{F}_2[\mathbf{L}]$. Finally, the implementation cost of $\mathbf{H}$ is less than 392 XOR for 8-bit input. But we could not get EGFS matrices under the stated conditions. Therefore, we increased the number of $8\times 8$ EGFS matrices.

Consider $8\times 8$ EGFS matrices $\mathbf{E}_i$ with $1\le i\le 5$, given in (27). The structures of $\mathbf{E}_i$'s are the same, since $\mathbf{E}_i$'s are constructed from two permutations $\mathbf{p}_1=\{4,3,2,1\}$ and

$\mathbf{p}_2 = \{3, 2, 1, 4\}$. Moreover, by using Appendix A, it can be checked that EGFS matrices $\mathbf{E}_i$ with $1 \leq i \leq 5$ are 6-primitive matrices over $\mathbb{R}$.

$$\mathbf{c}_1^{(1,m)} = \cdots = \mathbf{c}_4^{(1,m)} = \begin{pmatrix} \mathbf{1} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}, \qquad \mathbf{c}_1^{(2,m)} = \cdots = \mathbf{c}_4^{(2,m)} = \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{1} & \mathbf{0} \end{pmatrix},$$

$$\mathbf{E}_1 = \left(\begin{array}{cc|cc|cc|cc} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} \end{array}\right)$$

$$\mathbf{c}_1^{(1,m)} = \cdots = \mathbf{c}_4^{(1,m)} = \begin{pmatrix} \mathbf{1} & \mathbf{L} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}, \qquad \mathbf{c}_1^{(1,m)} = \cdots = \mathbf{c}_4^{(1,m)} = \begin{pmatrix} \mathbf{L} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} \end{pmatrix},$$

$$\mathbf{c}_1^{(2,m)} = \cdots = \mathbf{c}_4^{(2,m)} = \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{1} & \mathbf{0} \end{pmatrix}, \qquad \mathbf{c}_1^{(2,m)} = \cdots = \mathbf{c}_4^{(2,m)} = \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{1} & \mathbf{0} \end{pmatrix},$$

$$\mathbf{E}_2 = \left(\begin{array}{cc|cc|cc|cc} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{L} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{L} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{L} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{1} & \mathbf{L} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} \end{array}\right), \qquad \mathbf{E}_3 = \left(\begin{array}{cc|cc|cc|cc} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{L} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{L} & \mathbf{1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{L} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{L} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} \end{array}\right)$$

$$\mathbf{c}_1^{(1,m)} = \cdots = \mathbf{c}_4^{(1,m)} = \begin{pmatrix} \mathbf{1} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}, \qquad \mathbf{c}_1^{(1,m)} = \cdots = \mathbf{c}_4^{(1,m)} = \begin{pmatrix} \mathbf{1} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} \end{pmatrix},$$

$$\mathbf{c}_1^{(2,m)} = \cdots = \mathbf{c}_4^{(2,m)} = \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{L}^{-1} & \mathbf{0} \end{pmatrix}, \qquad \mathbf{c}_1^{(2,m)} = \cdots = \mathbf{c}_4^{(2,m)} = \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{L} & \mathbf{0} \end{pmatrix},$$

$$\mathbf{E}_4 = \left(\begin{array}{cc|cc|cc|cc} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{L^{-1}} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{L^{-1}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{L^{-1}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{L^{-1}} & \mathbf{0} \end{array}\right), \qquad \mathbf{E}_5 = \left(\begin{array}{cc|cc|cc|cc} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{L} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{L} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{L} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{L} & \mathbf{0} \end{array}\right)$$

(27)

Now based on the EGFS matrices $\mathbf{E}_i$ with $1 \leq i \leq 5$, we propose an MDS matrix $\mathbf{H}$ over $\mathbb{F}_2[\mathbf{L}]$ such that $\mathbf{H}$ is constructed from seven EGFS matrices.

$$\begin{aligned} \mathbf{H} &= \mathbf{E}_1\mathbf{E}_2\mathbf{E}_1\mathbf{E}_3\mathbf{E}_4\mathbf{E}_5\mathbf{E}_1 \\ &= \begin{pmatrix} \mathbf{L^2+L+1} & \mathbf{L^2+L+1} & \mathbf{L+L^{-1}} & \mathbf{L^2+L} \\ \mathbf{L^2+L+L^{-1}+1} & \mathbf{L+1} & \mathbf{L^2+L+L^{-1}} & \mathbf{L^2+L^{-1}} \\ \mathbf{L+L^{-1}} & \mathbf{L^2+L} & \mathbf{L^2} & \mathbf{L^2+L+L^{-1}} \\ \mathbf{L^2+L+L^{-1}} & \mathbf{L^2+L^{-1}} & \mathbf{L^3+L+1} & \mathbf{L^3+L} \\ \mathbf{L^2} & \mathbf{L^2+L+L^{-1}} & \mathbf{L^3+L^{-1}+1} & \mathbf{L^3+L+L^{-1}} \\ \mathbf{L^3+L+1} & \mathbf{L^3+L} & \mathbf{L^2} & \mathbf{L^2+1} \\ \mathbf{L^3+L^{-1}+1} & \mathbf{L^3+L+L^{-1}} & \mathbf{L^2+L+1} & \mathbf{L^2+L+1} \\ \mathbf{L^2} & \mathbf{L^2+1} & \mathbf{L^2+L+L^{-1}+1} & \mathbf{L+1} \end{pmatrix} \\ &\quad \begin{pmatrix} \mathbf{L^2} & \mathbf{L^2+L+L^{-1}} & \mathbf{L^3+L^{-1}+1} & \mathbf{L^3+L+L^{-1}} \\ \mathbf{L^3+L+1} & \mathbf{L^3+L} & \mathbf{L^2} & \mathbf{L^2+1} \\ \mathbf{L^3+L^{-1}+1} & \mathbf{L^3+L+L^{-1}} & \mathbf{L^2+L+1} & \mathbf{L^2+L+1} \\ \mathbf{L^2} & \mathbf{L^2+1} & \mathbf{L^2+L+L^{-1}+1} & \mathbf{L+1} \\ \mathbf{L^2+L+1} & \mathbf{L^2+L+1} & \mathbf{L+L^{-1}} & \mathbf{L^2+L} \\ \mathbf{L^2+L+L^{-1}+1} & \mathbf{L+1} & \mathbf{L^2+L+L^{-1}} & \mathbf{L^2+L^{-1}} \\ \mathbf{L+L^{-1}} & \mathbf{L^2+L} & \mathbf{L^2} & \mathbf{L^2+L+L^{-1}} \\ \mathbf{L^2+L+L^{-1}} & \mathbf{L^2+L^{-1}} & \mathbf{L^3+L+1} & \mathbf{L^3+L} \end{pmatrix}. \end{aligned} \tag{28}$$

It follows from (28) that the implementation cost of matrix $\mathbf{H}$ for $m$-bit input is equal to

$$\overbrace{(4m)}^{\#\mathbf{E}_1}+\overbrace{(4m+4(\#\mathbf{L}))}^{\#\mathbf{E}_2}+\overbrace{(4m)}^{\#\mathbf{E}_1}+\overbrace{(4m+4(\#\mathbf{L}))}^{\#\mathbf{E}_3}+\overbrace{(4m+4(\#\mathbf{L}^{-1}))}^{\#\mathbf{E}_4}+\overbrace{(4m+4(\#\mathbf{L}))}^{\#\mathbf{E}_5}$$
$$+\overbrace{(4m)}^{\#\mathbf{E}_1}=28m+12(\#\mathbf{L})+4(\#\mathbf{L}^{-1}). \tag{29}$$

The base set of subdeterminants of $\mathbf{H}$ has 285 elements that are provided in Appendix C. In this base set, there are all irreducible polynomials of degrees 4 and 8, except for the primitive polynomial `0x1E7`. It can be checked that the given elements in the Appendix C, are non-singular matrices over $\mathbb{F}_2$ by applying the following non-singular $8\times 8$ binary matrices $\mathbf{L}_i$ with $1\leq i\leq 34$.

$$\begin{array}{ll}
\mathbf{L}_1=[[1,8],[1,3,7],[2],[3],[4],[5],[6],[7]], & \mathbf{L}_2=[[1,8],[1,3],[2,8],[3],[4],[5],[6],[7]],\\
\mathbf{L}_3=[[1,8],[1,7],[2,4],[3],[4],[5],[6],[7]], & \mathbf{L}_4=[[1,8],[1,7],[2],[3,5],[4],[5],[6],[7]],\\
\mathbf{L}_5=[[1,8],[1,7],[2],[3],[4,6],[5],[6],[7]], & \mathbf{L}_6=[[1,8],[1,7],[2],[3],[4],[5,7],[6],[7]],\\
\mathbf{L}_7=[[1,8],[1,7],[2],[3],[4],[5],[6,8],[7]], & \mathbf{L}_8=[[1,8],[1],[2,4,8],[3],[4],[5],[6],[7]],\\
\mathbf{L}_9=[[1,8],[1],[2,8],[3,5],[4],[5],[6],[7]], & \mathbf{L}_{10}=[[1,8],[1],[2,8],[3],[4,6],[5],[6],[7]],\\
\mathbf{L}_{11}=[[1,8],[1],[2,8],[3],[4],[5,7],[6],[7]], & \mathbf{L}_{12}=[[1,8],[1],[2,8],[3],[4],[5],[6,8],[7]],\\
\mathbf{L}_{13}=[[2,6,8],[1],[2],[3],[4],[5],[6,7],[7]], & \mathbf{L}_{14}=[[2,6,8],[1],[2],[3],[4],[5],[6],[7,8]],\\
\mathbf{L}_{15}=[[2,8],[1,7],[2],[3],[4],[5],[6],[7,8]], & \mathbf{L}_{16}=[[6,8],[1,3],[2],[3],[4],[5],[6,7],[7]],\\
\mathbf{L}_{17}=[[6,8],[1,3],[2],[3],[4],[5],[6],[7,8]], & \mathbf{L}_{18}=[[6,8],[1],[2,4],[3],[4],[5],[6,7],[7]],\\
\mathbf{L}_{19}=[[6,8],[1],[2,4],[3],[4],[5],[6],[7,8]], & \mathbf{L}_{20}=[[6,8],[1],[2],[3,5],[4],[5],[6,7],[7]],\\
\mathbf{L}_{21}=[[6,8],[1],[2],[3,5],[4],[5],[6],[7,8]], & \mathbf{L}_{22}=[[6,8],[1],[2],[3],[4,6],[5],[6,7],[7]],\\
\mathbf{L}_{23}=[[6,8],[1],[2],[3],[4,6],[5],[6],[7,8]], & \mathbf{L}_{24}=[[6,8],[1],[2],[3],[4],[5,7],[6],[7,8]],\\
\mathbf{L}_{25}=[[8],[1,2],[2,4,8],[3],[4],[5],[6],[7]], & \mathbf{L}_{26}=[[8],[1,2],[2,8],[3,5],[4],[5],[6],[7]],\\
\mathbf{L}_{27}=[[8],[1,2],[2,8],[3],[4,6],[5],[6],[7]], & \mathbf{L}_{28}=[[8],[1,2],[2,8],[3],[4],[5,7],[6],[7]],\\
\mathbf{L}_{29}=[[8],[1,2],[2,8],[3],[4],[5],[6,8],[7]], & \mathbf{L}_{30}=[[8],[1,3,7],[2],[3],[4],[5],[6],[7,8]],\\
\mathbf{L}_{31}=[[8],[1,7],[2,4],[3],[4],[5],[6],[7,8]], & \mathbf{L}_{32}=[[8],[1,7],[2],[3,5],[4],[5],[6],[7,8]],\\
\mathbf{L}_{33}=[[8],[1,7],[2],[3],[4,6],[5],[6],[7,8]], & \mathbf{L}_{34}=[[8],[1,7],[2],[3],[4],[5,7],[6],[7,8]].
\end{array} \tag{30}$$

Moreover, the implementation cost of $\mathbf{L}_i$'s, given in (30), is three XOR. Although, the implementation cost of $\mathbf{L}_i^{-1}$ with $1\leq i\leq 34$ are not three XOR, the inverse of $\mathbf{L}_i$'s can be implemented with three XOR. For instance, consider $\mathbf{x}=[x_1,x_2,\cdots,x_8]$. Then we have

$$\mathbf{L}_2^{-1}\cdot\mathbf{x}^T=[x_2+x_4,x_1+x_2+x_3+x_4,x_4,x_5,x_6,x_7,x_8,x_1+x_2+x_4]^T,$$
$$u_1=x_2+x_4,\ u_2=u_1+x_1,\ u_3=u_2+x_3.$$

Hence, using $\mathbf{L}_i$ with $1\leq i\leq 34$ and relation (29), $\mathbf{H}$ is implemented by $28\times 8+4\times 3+12\times 3=272$ XOR for 8-bit input. Now consider $\alpha$ is a root of `0x1E7`. Then it can be verified that the implementation cost of $\alpha$ and $\alpha^{-1}$ are five XOR. Therefore, $\mathbf{H}$ is implemented with $28\times 8+4\times 5+12\times 5=304$ XOR over $\mathbb{F}_{2^8}/$`0x1E7`.

Moreover, let $\mathbf{L}$ be an $2^k\times 2^k$ with $k>2$ companion binary matrix such that the characteristic polynomial of $\mathbf{L}$ over $\mathbb{F}_2$ is $\left(x^8+x^7+x^6+x^5+x^2+x+1\right)^{2^{k-3}}$. It can be proved that using $\mathbf{L}$ the elements of the base set of subdeterminants of $\mathbf{H}$, given in Appendix C, are non-singular matrices over $\mathbb{F}_2$. Furthermore, the implementation cost of $\mathbf{L}$ and $\mathbf{L}^{-1}$ are five XOR. Hence, by applying $\mathbf{L}$ and relation (29) the implementation cost of $\mathbf{H}$ is $28\times 2^k+80$ XOR for $2^k$-bit input with $k>2$.

Although we proposed an $8\times 8$ lightweight MDS matrix, we strongly believed that by applying binary linear functions to EGFS matrices it is possible to get an $8\times 8$ MDS matrix with the implementation cost less than 272 XOR for 8-bit input. Actually, the various structure of EGFS matrices give the possibility to get a better result.

# 7 Conclusion

This paper proposes a construction heuristic method to design MDS matrices with low hardware implementation cost based on the generalized Feistel structures. Feistel-based structures such as GFS structure are suitable choices to construct MDS matrices, since their inverses can be implemented with simplicity. First of all, using GFS structure, some types of sparse matrices, called primitive GFS matrices, are proposed. Then, by applying binary linear functions to primitive GFS matrices we proposed $4 \times 4$ and $8 \times 8$ MDS matrices. The proposed $4 \times 4$ and $8 \times 8$ matrices and their inverse are implemented with 68 and 280 XOR for 8-bit input, respectively. Next, using an extension of primitive GFS matrices we defined another type of sparse matrices called EGFS matrices. Then based on the EGFS matrices, $4 \times 4$, $6 \times 6$ and $8 \times 8$ lightweight MDS matrices are implemented with 67, 158 and 272 XOR for 8-bit input, respectively.

One of the important features of this work is that the proposed MDS matrices are not only efficient by hardware terminology, but also are suitable by software perspective. In fact, proposed $4 \times 4$, $6 \times 6$ and $8 \times 8$ lightweight MDS matrices are constructed from 3, 7 and 16 binary linear functions. A summary of results of this paper is presented in Table 2.

**Table 2:** A summary of results of this paper.

| Iteration | Implementation Cost | Total Cost | Inverse Cost | Depth | Fig. |
|---|---|---|---|---|---|
| Lightweight $4 \times 4$ MDS Matrices for 4-bit | | | | | |
| 4 Round | 8 $\text{XOR}_{\text{4-bit}}$, 4 **L** | 36 $\text{XOR}_{\text{1-bit}}$ | 36 $\text{XOR}_{\text{1-bit}}$ | 6 | 1 |
| 4 Round | 8 $\text{XOR}_{\text{4-bit}}$, 3 **L** | 35 $\text{XOR}_{\text{1-bit}}$ | 36 $\text{XOR}_{\text{1-bit}}$ | 5 | 2 |
| Lightweight $4 \times 4$ MDS Matrices for 8-bit | | | | | |
| 4 Round | 8 $\text{XOR}_{\text{8-bit}}$, 4 **L** | 68 $\text{XOR}_{\text{1-bit}}$ | 68 $\text{XOR}_{\text{1-bit}}$ | 6 | 1 |
| 4 Round | 8 $\text{XOR}_{\text{8-bit}}$, 3 **L** | 67 $\text{XOR}_{\text{1-bit}}$ | 68 $\text{XOR}_{\text{1-bit}}$ | 5 | 2 |
| Lightweight $6 \times 6$ MDS Matrices for 8-bit | | | | | |
| 6 Round | 18 $\text{XOR}_{\text{8-bit}}$, 7 **L** | 158 $\text{XOR}_{\text{1-bit}}$ | 160 $\text{XOR}_{\text{1-bit}}$ | 8 | 3 |
| Lightweight $8 \times 8$ MDS Matrices for 8-bit | | | | | |
| **7** Round | 28 $\text{XOR}_{\text{8-bit}}$, 20 **L** | 280 $\text{XOR}_{\text{1-bit}}$ | 280 $\text{XOR}_{\text{1-bit}}$ | 11 | 4 |
| **7** Round | 28 $\text{XOR}_{\text{8-bit}}$, 16 **L** | 272 $\text{XOR}_{\text{1-bit}}$ | 296 $\text{XOR}_{\text{1-bit}}$ | 9 | 5 |

**Acknowledgments**

The authors would like to thank the anonymous referees for their constructive comments.

## References


[Aug14] D. Augot and M. Finiasz. Direct construction of recursive MDS diffusion layers using shortened BCH codes. In *FSE*, volume 8540, pages 3–17. Springer, 2014. doi:10.1007/978-3-662-46706-0_1.

[Bei16] C. Beierle, T. Kranz, and G. Leander. Lightweight multiplicationin GF($2^n$) with applications to MDS matrices. In *CRYPTO*, volume 9814, page 625–653. Springer, 2016. doi:10.1007/978-3-662-53018-4_23.

[Ber13] T. Berger. Construction of recursive MDS diffusion layers from Gabidulin codes. In *INDOCRYPT*, volume 8250, pages 274–285. Springer, 2013. doi:10.1007/978-3-319-03515-4_18.

[Bla99] M. Blaum and R. Roth. On lowest density mds codes. *IEEE Trans. Inform. Theory*, 45(1):46–59, 1999. doi:10.1109/18.746771.

[Duv18] S. Duval and G. Leurent. MDS Matrices with Lightweight Circuits. In *FSE*, volume 2018, pages 48–78. Springer, 2018. doi:10.13154/tosc.v2018.i2.48-78.

[Guo11] J. Guo, T. Peyrin, and A. Poschmann. The PHOTON family of lightweight hash functions. In *CRYPTO*, volume 684, page 222–239. Springer, 2011. doi:10.1007/978-3-642-22792-9_13.

[Gup13] K. Gupta and I. Ray. On constructions of involutory MDS matrices. In *AFRICACRYPT*, volume 7918, pages 43–60. Springer, 2013. doi:10.1007/978-3-642-38553-7_3.

[Hor13] R. Horn and C. Johnson. *Matrix Analysis*. Cambridge U.P, 2013.

[Kra17] H. Kranz, G. Leander, K. Stoffelen, and F. Wiemer. Shorter Linear Straight-Line Programs for MDS Matrices. In *FSE*, volume 2017, pages 188–211. Springer, 2017. doi:10.13154/tosc.v2017.i4.188-211.

[Saj12] M. Sajadieh, M. Dakhilalian, H. Mala, and P. Sepehrdad. Efficient diffusion layers for block ciphers and hash functions. In *FSE*, volume 7549, pages 385–401. Springer, 2012. doi:10.1007/978-3-642-34047-5_22.

[Sar16] S. Sarkar and H. Syed. Lightweight Diffusion Layer: Importance of Toeplitz Matrices. In *FSE*, volume 2016, pages 95–113. Springer, 2016. doi:10.13154/tosc.v2016.i1.95-113.

[Shi11] K. Shibutani. On the Diffusion of Generalized Feistel Structures Regarding Differential and Linear Cryptanalysis. In *SAC*, volume 6544, pages 211–228. Springer, 2011. doi:10.1007/978-3-642-19574-7_15.

[Sim15] S. M. Sim, K. Khoo, F. Oggier, and T. Peyrin. Lightweight MDS Involution Matrices. In *FSE*, volume 9054, pages 471–493. Springer, 2015. doi:10.1007/978-3-662-48116-5_23.

[Toh18] D. Toh, J. Teo, K. Khoo, and S. Sim. Lightweight MDS Serial-Type Matrices with Minimal Fixed XOR Count. In *AFRICACRYPT*, volume 10831, pages 51–71. Springer, 2018. doi:10.1007/978-3-319-89339-6_4.

[Wu12] S. Wu, M. Wang, and W. Wu. Recursive Diffusion Layers for (Lightweight) Block Ciphers and Hash Functions. In *SAC*, volume 7707, pages 355–371. Springer, 2012. doi:10.1007/978-3-642-35999-6_23.

[You97] A. Youssef, S. Mister, and S. Tavares. On the design of linear transformations for substitution permutation encryption networks. In *SAC*, pages 40–48, 1997.

# Appendix A

| Case | $\mathbf{p}_1$ | $\mathbf{p}_2$ | Order |
|---|---|---|---|
| 1 | $\{1,2,3,4\}$ | $\{2,1,4,3\}$ | $\infty$ |
| 2 | $\{1,2,3,4\}$ | $\{2,3,4,1\}$ | 8 |
| 3 | $\{1,2,3,4\}$ | $\{2,4,1,3\}$ | 8 |
| 4 | $\{1,2,3,4\}$ | $\{3,1,4,2\}$ | 8 |
| 5 | $\{1,2,3,4\}$ | $\{3,4,1,2\}$ | $\infty$ |
| 6 | $\{1,2,3,4\}$ | $\{3,4,2,1\}$ | 8 |
| 7 | $\{1,2,3,4\}$ | $\{4,1,2,3\}$ | 8 |
| 8 | $\{1,2,3,4\}$ | $\{4,3,1,2\}$ | 8 |
| 9 | $\{1,2,3,4\}$ | $\{4,3,2,1\}$ | $\infty$ |
| 10 | $\{1,2,4,3\}$ | $\{2,1,3,4\}$ | $\infty$ |
| 11 | $\{1,2,4,3\}$ | $\{2,3,1,4\}$ | 7 |
| 12 | $\{1,2,4,3\}$ | $\{2,4,3,1\}$ | 7 |
| 13 | $\{1,2,4,3\}$ | $\{3,1,2,4\}$ | 7 |
| 14 | $\{1,2,4,3\}$ | $\{3,4,1,2\}$ | 6 |
| 15 | $\{1,2,4,3\}$ | $\{3,4,2,1\}$ | 7 |
| 16 | $\{1,2,4,3\}$ | $\{4,1,3,2\}$ | 7 |
| 17 | $\{1,2,4,3\}$ | $\{4,3,1,2\}$ | 7 |
| 18 | $\{1,2,4,3\}$ | $\{4,3,2,1\}$ | 6 |
| 19 | $\{1,3,2,4\}$ | $\{2,1,4,3\}$ | 6 |
| 20 | $\{1,3,2,4\}$ | $\{2,4,1,3\}$ | 7 |
| 21 | $\{1,3,2,4\}$ | $\{2,4,3,1\}$ | 7 |
| 22 | $\{1,3,2,4\}$ | $\{3,1,4,2\}$ | 7 |
| 23 | $\{1,3,2,4\}$ | $\{3,2,4,1\}$ | 7 |
| 24 | $\{1,3,2,4\}$ | $\{3,4,1,2\}$ | 6 |
| 25 | $\{1,3,2,4\}$ | $\{4,1,3,2\}$ | 7 |
| 26 | $\{1,3,2,4\}$ | $\{4,2,1,3\}$ | 7 |
| 27 | $\{1,3,2,4\}$ | $\{4,2,3,1\}$ | $\infty$ |
| 28 | $\{1,3,4,2\}$ | $\{2,1,3,4\}$ | 7 |
| 29 | $\{1,3,4,2\}$ | $\{2,4,1,3\}$ | 6 |
| 30 | $\{1,3,4,2\}$ | $\{2,4,3,1\}$ | 6 |
| 31 | $\{1,3,4,2\}$ | $\{3,1,2,4\}$ | 6 |
| 32 | $\{1,3,4,2\}$ | $\{3,2,1,4\}$ | 7 |
| 33 | $\{1,3,4,2\}$ | $\{3,4,2,1\}$ | 6 |
| 34 | $\{1,3,4,2\}$ | $\{4,1,2,3\}$ | 6 |
| 35 | $\{1,3,4,2\}$ | $\{4,2,1,3\}$ | 6 |
| 36 | $\{1,3,4,2\}$ | $\{4,2,3,1\}$ | 7 |
| 37 | $\{1,4,2,3\}$ | $\{2,1,3,4\}$ | 7 |
| 38 | $\{1,4,2,3\}$ | $\{2,3,1,4\}$ | 6 |
| 39 | $\{1,4,2,3\}$ | $\{2,3,4,1\}$ | 6 |
| 40 | $\{1,4,2,3\}$ | $\{3,1,4,2\}$ | 6 |
| 41 | $\{1,4,2,3\}$ | $\{3,2,1,4\}$ | 7 |
| 42 | $\{1,4,2,3\}$ | $\{3,2,4,1\}$ | 6 |
| 43 | $\{1,4,2,3\}$ | $\{4,1,3,2\}$ | 6 |
| 44 | $\{1,4,2,3\}$ | $\{4,2,3,1\}$ | 7 |
| 45 | $\{1,4,2,3\}$ | $\{4,3,1,2\}$ | 6 |
| 46 | $\{1,4,3,2\}$ | $\{2,1,4,3\}$ | 6 |
| 47 | $\{1,4,3,2\}$ | $\{2,3,1,4\}$ | 7 |
| 48 | $\{1,4,3,2\}$ | $\{2,3,4,1\}$ | 7 |
| 49 | $\{1,4,3,2\}$ | $\{3,1,2,4\}$ | 7 |
| 50 | $\{1,4,3,2\}$ | $\{3,2,1,4\}$ | $\infty$ |
| 51 | $\{1,4,3,2\}$ | $\{3,2,4,1\}$ | 7 |
| 52 | $\{1,4,3,2\}$ | $\{4,1,2,3\}$ | 7 |
| 53 | $\{1,4,3,2\}$ | $\{4,2,1,3\}$ | 7 |
| 54 | $\{1,4,3,2\}$ | $\{4,3,2,1\}$ | 6 |

| Case | $\mathbf{p}_1$ | $\mathbf{p}_2$ | Order |
|---|---|---|---|
| 55 | $\{2,1,3,4\}$ | $\{1,2,4,3\}$ | $\infty$ |
| 56 | $\{2,1,3,4\}$ | $\{1,3,4,2\}$ | 7 |
| 57 | $\{2,1,3,4\}$ | $\{1,4,2,3\}$ | 7 |
| 58 | $\{2,1,3,4\}$ | $\{3,2,4,1\}$ | 7 |
| 59 | $\{2,1,3,4\}$ | $\{3,4,1,2\}$ | 6 |
| 60 | $\{2,1,3,4\}$ | $\{3,4,2,1\}$ | 7 |
| 61 | $\{2,1,3,4\}$ | $\{4,2,1,3\}$ | 7 |
| 62 | $\{2,1,3,4\}$ | $\{4,3,1,2\}$ | 7 |
| 63 | $\{2,1,3,4\}$ | $\{4,3,2,1\}$ | 6 |
| 64 | $\{2,1,4,3\}$ | $\{1,2,3,4\}$ | $\infty$ |
| 65 | $\{2,1,4,3\}$ | $\{1,3,2,4\}$ | 6 |
| 66 | $\{2,1,4,3\}$ | $\{1,4,3,2\}$ | 6 |
| 67 | $\{2,1,4,3\}$ | $\{3,2,1,4\}$ | 6 |
| 68 | $\{2,1,4,3\}$ | $\{3,4,1,2\}$ | $\infty$ |
| 69 | $\{2,1,4,3\}$ | $\{3,4,2,1\}$ | 8 |
| 70 | $\{2,1,4,3\}$ | $\{4,2,3,1\}$ | 6 |
| 71 | $\{2,1,4,3\}$ | $\{4,3,1,2\}$ | 8 |
| 72 | $\{2,1,4,3\}$ | $\{4,3,2,1\}$ | $\infty$ |
| 73 | $\{2,3,1,4\}$ | $\{1,2,4,3\}$ | 7 |
| 74 | $\{2,3,1,4\}$ | $\{1,4,2,3\}$ | 6 |
| 75 | $\{2,3,1,4\}$ | $\{1,4,3,2\}$ | 7 |
| 76 | $\{2,3,1,4\}$ | $\{3,1,4,2\}$ | 6 |
| 77 | $\{2,3,1,4\}$ | $\{3,2,4,1\}$ | 6 |
| 78 | $\{2,3,1,4\}$ | $\{3,4,2,1\}$ | 6 |
| 79 | $\{2,3,1,4\}$ | $\{4,1,2,3\}$ | 6 |
| 80 | $\{2,3,1,4\}$ | $\{4,1,3,2\}$ | 6 |
| 81 | $\{2,3,1,4\}$ | $\{4,2,3,1\}$ | 7 |
| 82 | $\{2,3,4,1\}$ | $\{1,2,3,4\}$ | 8 |
| 83 | $\{2,3,4,1\}$ | $\{1,4,2,3\}$ | 7 |
| 84 | $\{2,3,4,1\}$ | $\{1,4,3,2\}$ | 7 |
| 85 | $\{2,3,4,1\}$ | $\{3,1,2,4\}$ | 7 |
| 86 | $\{2,3,4,1\}$ | $\{3,2,1,4\}$ | 7 |
| 87 | $\{2,3,4,1\}$ | $\{3,4,1,2\}$ | 8 |
| 88 | $\{2,3,4,1\}$ | $\{4,1,2,3\}$ | $\infty$ |
| 89 | $\{2,3,4,1\}$ | $\{4,1,3,2\}$ | 7 |
| 90 | $\{2,3,4,1\}$ | $\{4,2,1,3\}$ | 7 |
| 91 | $\{2,4,1,3\}$ | $\{1,2,3,4\}$ | 8 |
| 92 | $\{2,4,1,3\}$ | $\{1,3,2,4\}$ | 7 |
| 93 | $\{2,4,1,3\}$ | $\{1,3,4,2\}$ | 7 |
| 94 | $\{2,4,1,3\}$ | $\{3,1,2,4\}$ | 7 |
| 95 | $\{2,4,1,3\}$ | $\{3,1,4,2\}$ | $\infty$ |
| 96 | $\{2,4,1,3\}$ | $\{3,2,4,1\}$ | 7 |
| 97 | $\{2,4,1,3\}$ | $\{4,1,3,2\}$ | 7 |
| 98 | $\{2,4,1,3\}$ | $\{4,2,3,1\}$ | 7 |
| 99 | $\{2,4,1,3\}$ | $\{4,3,2,1\}$ | 8 |
| 100 | $\{2,4,3,1\}$ | $\{1,2,4,3\}$ | 7 |
| 101 | $\{2,4,3,1\}$ | $\{1,3,2,4\}$ | 7 |
| 102 | $\{2,4,3,1\}$ | $\{1,3,4,2\}$ | 6 |
| 103 | $\{2,4,3,1\}$ | $\{3,1,2,4\}$ | 6 |
| 104 | $\{2,4,3,1\}$ | $\{3,1,4,2\}$ | 6 |
| 105 | $\{2,4,3,1\}$ | $\{3,2,1,4\}$ | 7 |
| 106 | $\{2,4,3,1\}$ | $\{4,1,2,3\}$ | 6 |
| 107 | $\{2,4,3,1\}$ | $\{4,2,1,3\}$ | 6 |
| 108 | $\{2,4,3,1\}$ | $\{4,3,1,2\}$ | 6 |

| Case | $\mathbf{p}_1$ | $\mathbf{p}_2$ | Order |
|---|---|---|---|
| 109 | $\{3,1,2,4\}$ | $\{1,2,4,3\}$ | 7 |
| 110 | $\{3,1,2,4\}$ | $\{1,3,4,2\}$ | 6 |
| 111 | $\{3,1,2,4\}$ | $\{1,4,3,2\}$ | 7 |
| 112 | $\{3,1,2,4\}$ | $\{2,3,4,1\}$ | 6 |
| 113 | $\{3,1,2,4\}$ | $\{2,4,1,3\}$ | 6 |
| 114 | $\{3,1,2,4\}$ | $\{2,4,3,1\}$ | 6 |
| 115 | $\{3,1,2,4\}$ | $\{4,2,1,3\}$ | 6 |
| 116 | $\{3,1,2,4\}$ | $\{4,2,3,1\}$ | 7 |
| 117 | $\{3,1,2,4\}$ | $\{4,3,1,2\}$ | 6 |
| 118 | $\{3,1,4,2\}$ | $\{1,2,3,4\}$ | 8 |
| 119 | $\{3,1,4,2\}$ | $\{1,3,2,4\}$ | 7 |
| 120 | $\{3,1,4,2\}$ | $\{1,4,2,3\}$ | 7 |
| 121 | $\{3,1,4,2\}$ | $\{2,3,1,4\}$ | 7 |
| 122 | $\{3,1,4,2\}$ | $\{2,4,1,3\}$ | $\infty$ |
| 123 | $\{3,1,4,2\}$ | $\{2,4,3,1\}$ | 7 |
| 124 | $\{3,1,4,2\}$ | $\{4,2,1,3\}$ | 7 |
| 125 | $\{3,1,4,2\}$ | $\{4,2,3,1\}$ | 7 |
| 126 | $\{3,1,4,2\}$ | $\{4,3,2,1\}$ | 8 |
| 127 | $\{3,2,1,4\}$ | $\{1,3,4,2\}$ | 7 |
| 128 | $\{3,2,1,4\}$ | $\{1,4,2,3\}$ | 7 |
| 129 | $\{3,2,1,4\}$ | $\{1,4,3,2\}$ | $\infty$ |
| 130 | $\{3,2,1,4\}$ | $\{2,1,4,3\}$ | 6 |
| 131 | $\{3,2,1,4\}$ | $\{2,3,4,1\}$ | 7 |
| 132 | $\{3,2,1,4\}$ | $\{2,4,3,1\}$ | 7 |
| 133 | $\{3,2,1,4\}$ | $\{4,1,2,3\}$ | 7 |
| 134 | $\{3,2,1,4\}$ | $\{4,1,3,2\}$ | 7 |
| 135 | $\{3,2,1,4\}$ | $\{4,3,2,1\}$ | 6 |
| 136 | $\{3,2,4,1\}$ | $\{1,3,2,4\}$ | 7 |
| 137 | $\{3,2,4,1\}$ | $\{1,4,2,3\}$ | 6 |
| 138 | $\{3,2,4,1\}$ | $\{1,4,3,2\}$ | 7 |
| 139 | $\{3,2,4,1\}$ | $\{2,1,3,4\}$ | 7 |
| 140 | $\{3,2,4,1\}$ | $\{2,3,1,4\}$ | 6 |
| 141 | $\{3,2,4,1\}$ | $\{2,4,1,3\}$ | 6 |
| 142 | $\{3,2,4,1\}$ | $\{4,1,2,3\}$ | 6 |
| 143 | $\{3,2,4,1\}$ | $\{4,1,3,2\}$ | 6 |
| 144 | $\{3,2,4,1\}$ | $\{4,3,1,2\}$ | 6 |
| 145 | $\{3,4,1,2\}$ | $\{1,2,3,4\}$ | $\infty$ |
| 146 | $\{3,4,1,2\}$ | $\{1,2,4,3\}$ | 6 |
| 147 | $\{3,4,1,2\}$ | $\{1,3,2,4\}$ | 6 |
| 148 | $\{3,4,1,2\}$ | $\{2,1,3,4\}$ | 6 |
| 149 | $\{3,4,1,2\}$ | $\{2,1,4,3\}$ | $\infty$ |
| 150 | $\{3,4,1,2\}$ | $\{2,3,4,1\}$ | 8 |
| 151 | $\{3,4,1,2\}$ | $\{4,1,2,3\}$ | 8 |
| 152 | $\{3,4,1,2\}$ | $\{4,2,3,1\}$ | 6 |
| 153 | $\{3,4,1,2\}$ | $\{4,3,2,1\}$ | $\infty$ |
| 154 | $\{3,4,2,1\}$ | $\{1,2,3,4\}$ | 8 |
| 155 | $\{3,4,2,1\}$ | $\{1,2,4,3\}$ | 7 |
| 156 | $\{3,4,2,1\}$ | $\{1,3,4,2\}$ | 7 |
| 157 | $\{3,4,2,1\}$ | $\{2,1,3,4\}$ | 7 |
| 158 | $\{3,4,2,1\}$ | $\{2,1,4,3\}$ | 8 |
| 159 | $\{3,4,2,1\}$ | $\{2,3,1,4\}$ | 7 |
| 160 | $\{3,4,2,1\}$ | $\{4,1,3,2\}$ | 7 |
| 161 | $\{3,4,2,1\}$ | $\{4,2,1,3\}$ | 7 |
| 162 | $\{3,4,2,1\}$ | $\{4,3,1,2\}$ | $\infty$ |

| Case | $\mathbf{p}_1$ | $\mathbf{p}_2$ | Order |
|---|---|---|---|
| 163 | $\{4,1,2,3\}$ | $\{1,2,3,4\}$ | 8 |
| 164 | $\{4,1,2,3\}$ | $\{1,3,4,2\}$ | 7 |
| 165 | $\{4,1,2,3\}$ | $\{1,4,3,2\}$ | 7 |
| 166 | $\{4,1,2,3\}$ | $\{2,3,1,4\}$ | 7 |
| 167 | $\{4,1,2,3\}$ | $\{2,3,4,1\}$ | $\infty$ |
| 168 | $\{4,1,2,3\}$ | $\{2,4,3,1\}$ | 7 |
| 169 | $\{4,1,2,3\}$ | $\{3,2,1,4\}$ | 7 |
| 170 | $\{4,1,2,3\}$ | $\{3,2,4,1\}$ | 7 |
| 171 | $\{4,1,2,3\}$ | $\{3,4,1,2\}$ | 8 |
| 172 | $\{4,1,3,2\}$ | $\{1,2,4,3\}$ | 7 |
| 173 | $\{4,1,3,2\}$ | $\{1,3,2,4\}$ | 7 |
| 174 | $\{4,1,3,2\}$ | $\{1,4,2,3\}$ | 6 |
| 175 | $\{4,1,3,2\}$ | $\{2,3,1,4\}$ | 6 |
| 176 | $\{4,1,3,2\}$ | $\{2,3,4,1\}$ | 6 |
| 177 | $\{4,1,3,2\}$ | $\{2,4,1,3\}$ | 6 |
| 178 | $\{4,1,3,2\}$ | $\{3,2,1,4\}$ | 7 |
| 179 | $\{4,1,3,2\}$ | $\{3,2,4,1\}$ | 6 |
| 180 | $\{4,1,3,2\}$ | $\{3,4,2,1\}$ | 6 |
| 181 | $\{4,2,1,3\}$ | $\{1,3,2,4\}$ | 7 |
| 182 | $\{4,2,1,3\}$ | $\{1,3,4,2\}$ | 6 |
| 183 | $\{4,2,1,3\}$ | $\{1,4,3,2\}$ | 7 |
| 184 | $\{4,2,1,3\}$ | $\{2,1,3,4\}$ | 7 |
| 185 | $\{4,2,1,3\}$ | $\{2,3,4,1\}$ | 6 |
| 186 | $\{4,2,1,3\}$ | $\{2,4,3,1\}$ | 6 |
| 187 | $\{4,2,1,3\}$ | $\{3,1,2,4\}$ | 6 |
| 188 | $\{4,2,1,3\}$ | $\{3,1,4,2\}$ | 6 |
| 189 | $\{4,2,1,3\}$ | $\{3,4,2,1\}$ | 6 |
| 190 | $\{4,2,3,1\}$ | $\{1,3,2,4\}$ | $\infty$ |
| 191 | $\{4,2,3,1\}$ | $\{1,3,4,2\}$ | 7 |
| 192 | $\{4,2,3,1\}$ | $\{1,4,2,3\}$ | 7 |
| 193 | $\{4,2,3,1\}$ | $\{2,1,4,3\}$ | 6 |
| 194 | $\{4,2,3,1\}$ | $\{2,3,1,4\}$ | 7 |
| 195 | $\{4,2,3,1\}$ | $\{2,4,1,3\}$ | 7 |
| 196 | $\{4,2,3,1\}$ | $\{3,1,2,4\}$ | 7 |
| 197 | $\{4,2,3,1\}$ | $\{3,1,4,2\}$ | 7 |
| 198 | $\{4,2,3,1\}$ | $\{3,4,1,2\}$ | 6 |
| 199 | $\{4,3,1,2\}$ | $\{1,2,3,4\}$ | 8 |
| 200 | $\{4,3,1,2\}$ | $\{1,2,4,3\}$ | 7 |
| 201 | $\{4,3,1,2\}$ | $\{1,4,2,3\}$ | 7 |
| 202 | $\{4,3,1,2\}$ | $\{2,1,3,4\}$ | 7 |
| 203 | $\{4,3,1,2\}$ | $\{2,1,4,3\}$ | 8 |
| 204 | $\{4,3,1,2\}$ | $\{2,4,3,1\}$ | 7 |
| 205 | $\{4,3,1,2\}$ | $\{3,1,2,4\}$ | 7 |
| 206 | $\{4,3,1,2\}$ | $\{3,2,4,1\}$ | 7 |
| 207 | $\{4,3,1,2\}$ | $\{3,4,2,1\}$ | $\infty$ |
| 208 | $\{4,3,2,1\}$ | $\{1,2,3,4\}$ | $\infty$ |
| 209 | $\{4,3,2,1\}$ | $\{1,2,4,3\}$ | 6 |
| 210 | $\{4,3,2,1\}$ | $\{1,4,3,2\}$ | 6 |
| 211 | $\{4,3,2,1\}$ | $\{2,1,3,4\}$ | 6 |
| 212 | $\{4,3,2,1\}$ | $\{2,1,4,3\}$ | $\infty$ |
| 213 | $\{4,3,2,1\}$ | $\{2,4,1,3\}$ | 8 |
| 214 | $\{4,3,2,1\}$ | $\{3,1,4,2\}$ | 8 |
| 215 | $\{4,3,2,1\}$ | $\{3,2,1,4\}$ | 6 |
| 216 | $\{4,3,2,1\}$ | $\{3,4,1,2\}$ | $\infty$ |

# Appendix B

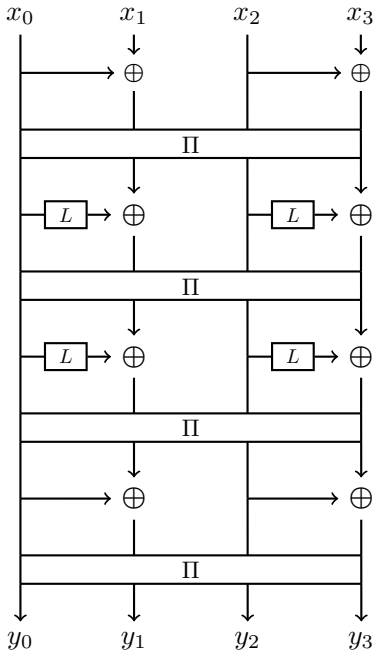


**Figure 1:** $4 \times 4$ MDS matrix with depth 6: $\mathbf{\Pi}(\mathbf{0},\mathbf{1},\mathbf{2},\mathbf{3}) = (\mathbf{3},\mathbf{0},\mathbf{1},\mathbf{2})$

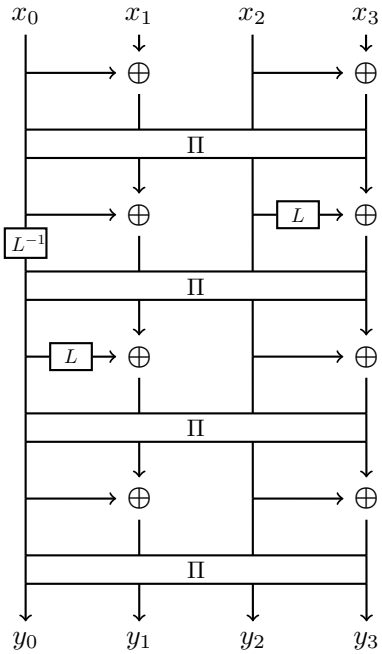


**Figure 2:** $4 \times 4$ MDS matrix with depth 5: $\mathbf{\Pi}(\mathbf{0},\mathbf{1},\mathbf{2},\mathbf{3}) = (\mathbf{3},\mathbf{0},\mathbf{1},\mathbf{2})$

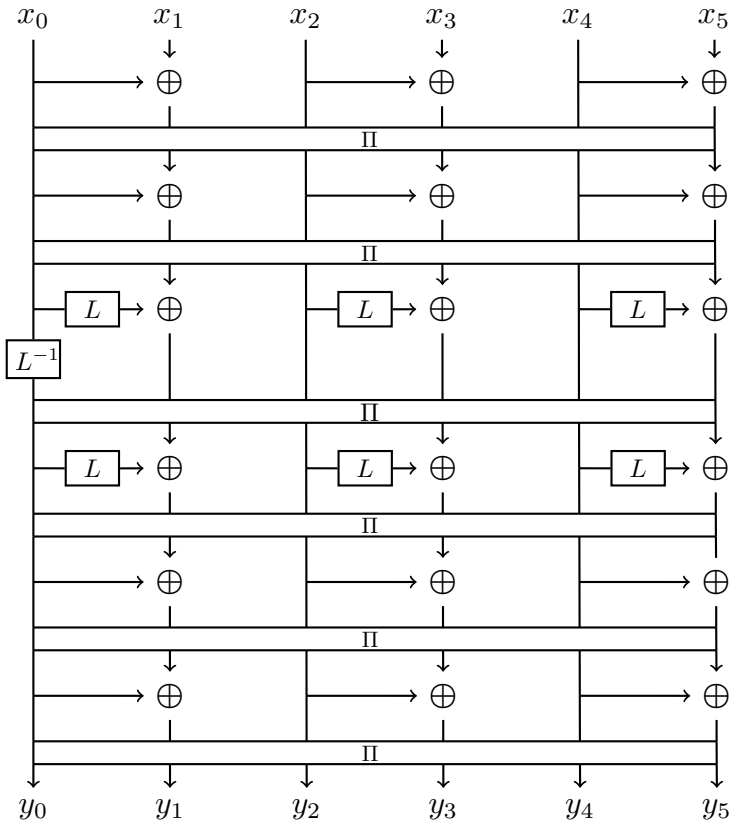


**Figure 3:** $6 \times 6$ MDS matrix with depth 8: $\mathbf{\Pi}(\mathbf{0},\mathbf{1},\mathbf{2},\mathbf{3},\mathbf{4},\mathbf{5}) = (\mathbf{5},\mathbf{2},\mathbf{1},\mathbf{4},\mathbf{3},\mathbf{0})$

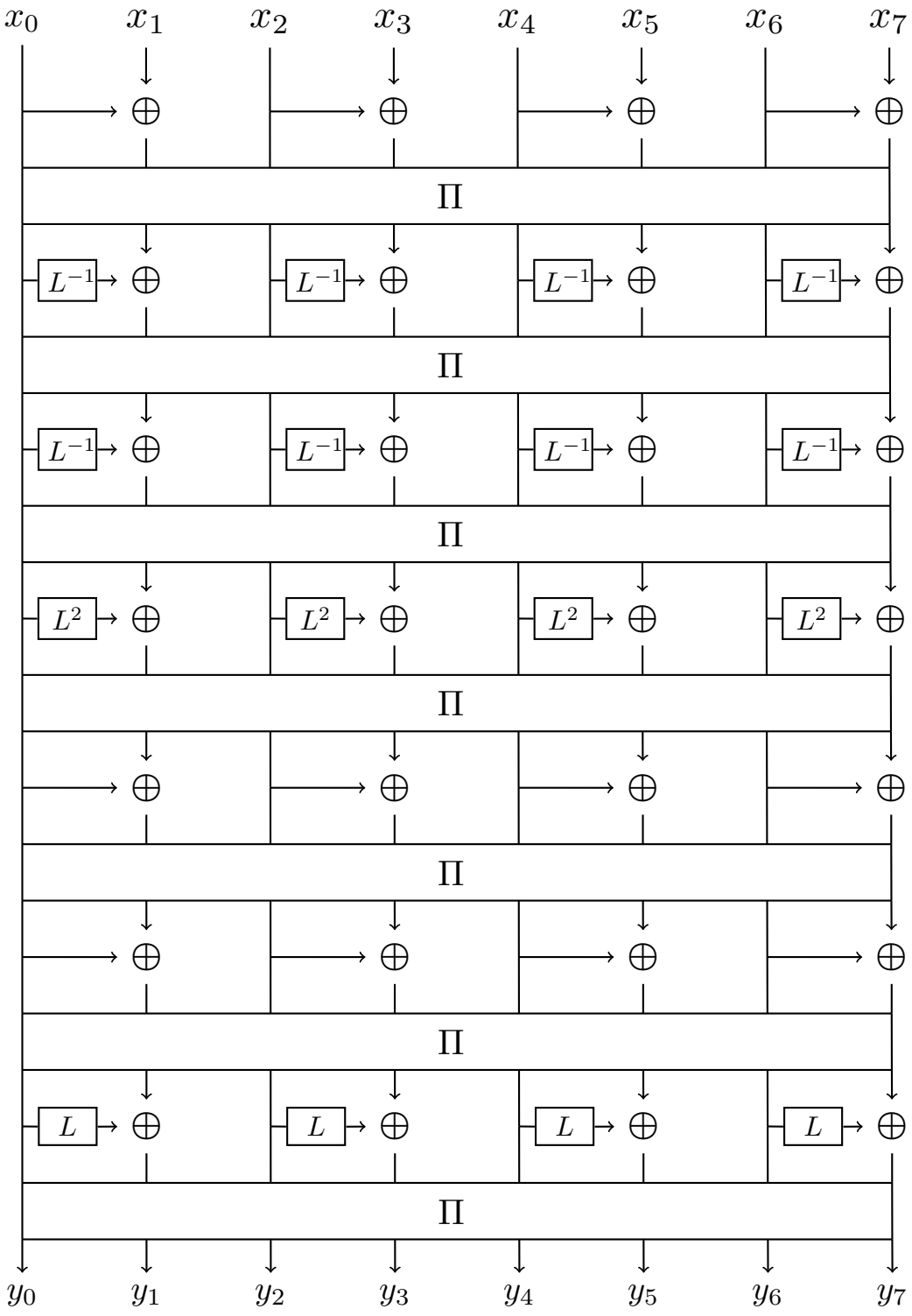


**Figure 4:** $8 \times 8$ MDS matrix with depth 11:
$\mathbf{\Pi}(\mathbf{0},\mathbf{1},\mathbf{2},\mathbf{3},\mathbf{4},\mathbf{5},\mathbf{6},\mathbf{7}) = (\mathbf{3},\mathbf{0},\mathbf{1},\mathbf{4},\mathbf{7},\mathbf{2},\mathbf{5},\mathbf{6})$

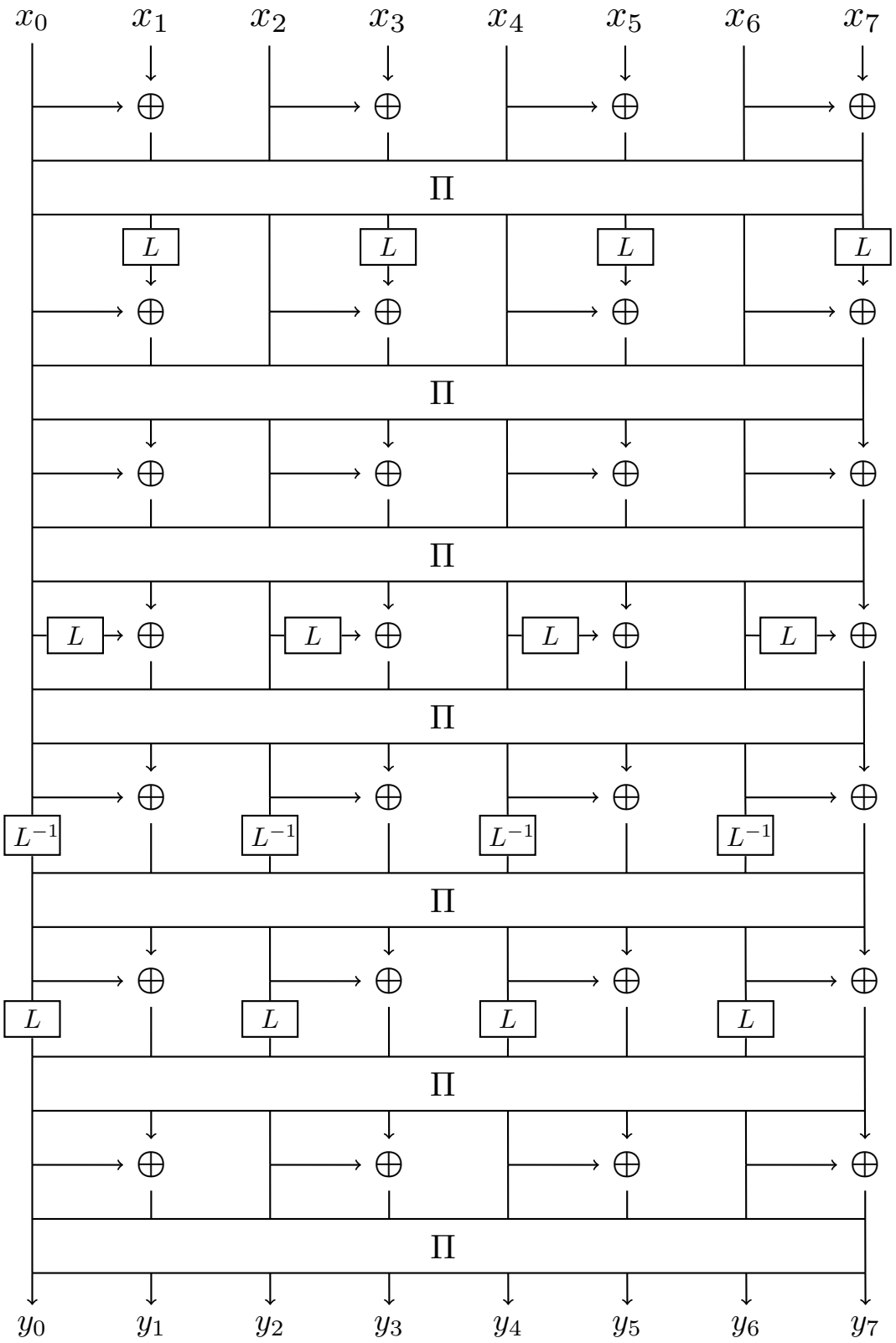


**Figure 5:** $8 \times 8$ MDS matrix with depth **9**:
$\mathbf{\Pi}(\mathbf{0}, \mathbf{1}, \mathbf{2}, \mathbf{3}, \mathbf{4}, \mathbf{5}, \mathbf{6}, \mathbf{7}) = (\mathbf{5}, \mathbf{6}, \mathbf{3}, \mathbf{4}, \mathbf{1}, \mathbf{2}, \mathbf{7}, \mathbf{0})$

## Appendix C

The base set of the matrix $\mathbf{H}$, given in (28), which are listed by hexadecimal values. For instance, we have $\texttt{0x14ABF} = L^{16} + L^{14} + L^{11} + L^9 + L^7 + L^5 + L^4 + L^3 + L^2 + L + 1$.

{0x2, 0x3, 0x7, 0xB, 0xD, 0x13, 0x19, 0x1F, 0x25, 0x29, 0x2F, 0x37, 0x3B, 0x3D, 0x43, 0x49, 0x57, 0x5B, 0x61, 0x67, 0x6D, 0x73, 0x75, 0x83, 0x89, 0x8F, 0x91, 0x9D, 0xA7, 0xAB, 0xB9, 0xBF, 0xC1, 0xCB, 0xD3, 0xD5, 0xE5, 0xEF, 0xF1, 0xF7, 0xFD, 0x11B, 0x11D, 0x12B, 0x12D, 0x139, 0x13F, 0x14D, 0x15F, 0x163, 0x165, 0x169, 0x171, 0x177, 0x17B, 0x187, 0x18B, 0x18D, 0x19F, 0x1A3, 0x1A9, 0x1B1, 0x1BD, 0x1C3, 0x1CF, 0x1D7, 0x1DD, 0x1F3, 0x1F5, 0x1F9, 0x203, 0x21B, 0x221, 0x22D, 0x233, 0x24B, 0x25F, 0x265, 0x269, 0x277, 0x27D, 0x287, 0x295, 0x299, 0x2A3, 0x2A5, 0x2B7, 0x2CF, 0x2DB, 0x2F5, 0x2F9, 0x313, 0x315, 0x31F, 0x331, 0x33B, 0x34F, 0x35B, 0x361, 0x36B, 0x36D, 0x373, 0x37F, 0x385, 0x3A1, 0x3B9, 0x3C7, 0x3CB, 0x3CD, 0x3D5, 0x3D9, 0x3E3, 0x3E9, 0x3FB, 0x409, 0x41B, 0x435, 0x447, 0x453, 0x465, 0x46F, 0x481, 0x4A9, 0x4C5, 0x4E7, 0x4F3, 0x4FF, 0x523, 0x53D, 0x543, 0x557, 0x58F, 0x59B, 0x5A1, 0x5AB, 0x5C7, 0x5F7, 0x615, 0x623, 0x631, 0x637, 0x64F, 0x65B, 0x679, 0x67F, 0x685, 0x689, 0x6A7, 0x6AD, 0x6B5, 0x6C1, 0x6CD, 0x711, 0x717, 0x71D, 0x721, 0x72B, 0x735, 0x755, 0x759, 0x77B, 0x77D, 0x781, 0x787, 0x7B1, 0x7C5, 0x7DB, 0x7F3, 0x7F9, 0x7FF, 0x805, 0x82D, 0x88D, 0x8A9, 0x8C3, 0x8CF, 0x8D1, 0x8E7, 0x93B, 0x949, 0x951, 0x973, 0x975, 0x9E5, 0x9EF, 0xA07, 0xA13, 0xA15, 0xA6D, 0xA79, 0xA7F, 0xAD5, 0xADF, 0xB11, 0xB33, 0xB3F, 0xB87, 0xB95, 0xBAF, 0xBBD, 0xBC9, 0xC0D, 0xC97, 0xCBF, 0xCC7, 0xD0F, 0xD1D, 0xD27, 0xD93, 0xDBB, 0xDC9, 0xDD7, 0xE27, 0xE2B, 0xE7B, 0xEA3, 0xEC9, 0xECF, 0xEF9, 0xF0B, 0xF19, 0xF6B, 0x1069, 0x1077, 0x10D1, 0x11EF, 0x1219, 0x13A9, 0x14B5, 0x154D, 0x1593, 0x15BB, 0x15C5, 0x16E7, 0x17FB, 0x1823, 0x1879, 0x197B, 0x19CF, 0x19F9, 0x1A69, 0x1BFD, 0x1C03, 0x1C27, 0x1CBB, 0x1CED, 0x1E3D, 0x1F11, 0x1F1B, 0x1FAF, 0x1FC3, 0x1FE1, 0x227F, 0x232B, 0x2429, 0x25BD, 0x2B2F, 0x2B97, 0x2F5F, 0x329F, 0x33E5, 0x3499, 0x3A61, 0x3FB5, 0x49E1, 0x4A17, 0x549F, 0x5585, 0x6A6B, 0x6D05, 0x7327, 0x74C7, 0x7BB9, 0x7CA3, 0xA6C7, 0xA7D1, 0xAF2F, 0xB08D, 0xB24F, 0xB909, 0xBB15, 0xD91B, 0xE05F, 0xEB97, 0x14ABF}