Design and analysis of MDS and Near-MDS Matrices and their application to lightweight cryptography

A thesis submitted to Indian Statistical Institute
in partial fulfillment of the thesis requirements for the degree of
Doctor of Philosophy in Computer Science

Author:
Susanta Samanta
susantas_r@isical.ac.in

Supervisor:
Prof. Kishan Chand Gupta
kishan@isical.ac.in

Applied Statistics Unit
Indian Statistical Institute
203, B. T. Road, Kolkata,
West Bengal, India - 700 108.

December, 2023

To my grandparents.

Acknowledgements

I would like to take this opportunity to express my sincere gratitude to all those who have contributed to the completion of my PhD thesis.

First and foremost, I am deeply grateful to my supervisor, Professor Kishan Chand Gupta, for his unwavering support, guidance, and mentorship throughout my research journey. Besides him, I am also grateful to Dr. Sumit Kumar Pandey for his valuable time, insightful comments, and expert guidance. Their extensive knowledge and expertise in the field of cryptography and finite fields have provided me with a comprehensive understanding of cryptography, expanding my horizons beyond what I knew prior to embarking on my PhD journey. The wisdom and experiences gained under their tutelage are invaluable, and I will always treasure them throughout my lifetime.

I am sincerely grateful to Prof. Bimal Kumar Roy, Prof. Rana Barua, Prof. Subhamoy Maitra, and Prof. Mridul Nandi for their invaluable help and guidance. I feel truly fortunate to have had Prof. Avishek Adhikari, Dr. Tapas Kumar Mondal, Dr. Anindita Basu, and Sujit Mondal as my teachers. Their guidance and mentorship have been instrumental in instilling confidence in me and propelling my career in the field of mathematical science.

I am thankful to the anonymous reviewers for their valuable comments. Additionally, I would like to express my gratitude to the members of our PhD and DSc committee.

I am also thankful to the former and current members of our research group, including Aniruddha, Jyotirmoy, Indranil-da, Samir-da, Avishek-da, Subhadip-da, Animesh, Chandranan, Suman, Rakesh, Subha, Siva, Anup, Kaushik-da, Prabal-da, Diptendu-da, Amit-da, Laltu-da, Mostaf-da, Sanjay-da, Avik-da, Nilanjan-da, Avijit-da, Srimanta-da and Butu-da with whom I have always had fruitful discussions and enjoyable times. I am grateful to all the members at ASU office and CSSC for their continuous support and for providing the necessary facilities whenever needed.

In addition to the research group, I would like to express my thanks to Sandip (Kumar Mondal), Anjan, Joginder, Pijush, Malay, Sandip (Moi), Parimal, Vola, and Tapu for their support and friendship throughout my journey.

I would like to express my heartfelt gratitude to my entire family for their unwavering support and guidance. In particular, I want to sincerely appreciate the contributions of my grandparents, Uday and Chaya, my parents, Sanatan and Chitra, and my uncle, Subrata. Their love and encouragement have played a crucial role

in both my personal and academic development. Last but not the least, I am grateful to my wife, Anurima, for her support, which has been a constant source of inspiration for me.

Susanta Samanta
Kolkata, December, 2023

Contents

Abstract

In this thesis, we focus on studying MDS and Near-MDS (NMDS) matrices and explore their construction in both recursive and nonrecursive settings. We present several theoretical results and analyze the hardware efficiency of MDS and NMDS matrix constructions. We begin by providing a comprehensive study of MDS matrices over finite fields. This study not only summarizes existing results but also reveals deep and nontrivial connections among various constructions of MDS matrices.

Next, we delve into the study of various sparse matrix structures for the construction of both MDS and NMDS matrices in recursive settings. Additionally, we explore various structures for the nonrecursive construction of NMDS matrices, including circulant and left-circulant matrices, as well as their generalizations such as Toeplitz and Hankel matrices. Whenever possible, we also make comparisons between the results of NMDS and MDS matrices.

Next, we present various techniques for direct constructions of MDS and NMDS matrices in both recursive and nonrecursive approaches. In the recursive approach, we derive recursive MDS and NMDS matrices from companion matrices, while direct constructions of nonrecursive MDS and NMDS matrices are obtained by using two generalized Vandermonde matrices. Furthermore, we propose a direct method for constructing involutory MDS and NMDS matrices.

Finally, we introduce FUTURE, a new SPN-based lightweight block cipher designed with minimal latency and low hardware implementation cost in mind. To achieve the best diffusion in the linear layer, FUTURE incorporates an MDS matrix in its round function. While the use of MDS matrices in lightweight block ciphers is typically avoided due to their high implementation cost. The MDS matrix in FUTURE is composed of four sparse matrices, striking a balance between diffusion property and implementation cost. In addition, FUTURE adopts a lightweight yet cryptographically significant Sbox, which is formed by combining four different Sboxes. By combining these design choices, FUTURE successfully combines lightweight implementation with the desirable properties of MDS matrices, offering an effective solution for designing lightweight block ciphers.

List of Work

1. Kishan Chand Gupta, Sumit Kumar Pandey, Indranil Ghosh Ray and **Susanta Samanta** "Cryptographically Significant MDS Matrices over Finite Fields: A Brief Survey and Some Generalized Results", Advances in Mathematics of Communication, 13(4): 779–843,
   DOI: https://doi.org/10.3934/amc.2019045.

2. Kishan Chand Gupta, Sumit Kumar Pandey and **Susanta Samanta** "A Few Negative Results on Constructions of MDS Matrices Using Low XOR Matrices", SPACE 2019: 195–213,
   DOI: https://doi.org/10.1007/978-3-030-35869-3_14.

3. Kishan Chand Gupta, Sumit Kumar Pandey and **Susanta Samanta** "Construction of Recursive MDS Matrices Using DLS Matrices", AFRICACRYPT 2022: 3–27,
   DOI: https://doi.org/10.1007/978-3-031-17433-9_1.

4. Kishan Chand Gupta, Sumit Kumar Pandey and **Susanta Samanta** "FUTURE: A Lightweight Block Cipher Using an Optimal Diffusion Matrix", AFRICACRYPT 2022: 28–52,
   DOI: https://doi.org/10.1007/978-3-031-17433-9_2.

5. Kishan Chand Gupta, Sumit Kumar Pandey and **Susanta Samanta** "On the Construction of Near-MDS Matrices", Cryptography and Communications, Aug, 2023,
   DOI: https://doi.org/10.1007/s12095-023-00667-x.

6. Kishan Chand Gupta, Sumit Kumar Pandey and **Susanta Samanta** "On the Direct Construction of MDS and Near-MDS Matrices", arXiv:2306.12848, https://arxiv.org/abs/2306.12848.

List of Figures

List of Tables

List of Abbreviations

Abbreviations	Expansion
AES	Advanced Encryption Standard
DLS	Diagonal-like sparse
DSI	Diagonal-Serial-Invertible
GDLS	Generalized DLS
GE	Gate equivalent
i.e.	That is
MDS	Maximum distance separable
MILP	Mixed-integer linear programming
NMDS	Near-MDS
XOR	Exclusive-OR

List of Symbols

In this thesis, various notations are employed, and an explanation of these commonly used symbols is provided here.

- $\mathbb{F}_q$: Finite field with q elements.
- $\text{Char}(R) = p$: The characteristic of the ring R is p.
- $\mathbb{F}_{2^4}/0\text{x}13$: The finite field $\mathbb{F}_{2^4}$ constructed by the polynomial $x^4 + x + 1$.
- $[n, k, d]$ code: Linear code of length n, dimension k, and minimum distance d.
- $\mathcal{C}^{\perp}$: Dual of the linear code $\mathcal{C}$.
- $M_{k\times n}(\mathbb{F}_q)$: The set of all matrices of size $k \times n$ over the field $\mathbb{F}_q$.
- $M_n(\mathbb{F}_q)$: The ring of all $n \times n$ matrices (square matrices of order n) over $\mathbb{F}_q$.
- $GL(n, \mathbb{F}_q)$: The general linear group consisting of nonsingular $n \times n$ matrices over $\mathbb{F}_q$.
- I_n: The identity matrix of $M_n(\mathbb{F}_q)$.
- $\det(A)$: The determinant of a matrix A.
- $(A)_{i,j}$: The (i, j)-th entry of the matrix A.
- $A_{row(i)}$: The i-th row of A.
- $A_{column(j)}$: The j-th column of A.
- A^r: The r-th power of the matrix A i.e. $A^r = \underbrace{A \times A \times \ldots \times A}_{\text{r times}}$.
- $|A|$: The number of nonzero entries in the matrix A.
- $|A| \leq |B|$: The number of nonzero elements in the matrix A is less than or equal to the number of nonzero elements in the matrix B.
- $A \leqq B$: $(A)_{i,j} \neq 0 \implies (B)_{i,j} \neq 0$.
- $diag(d_1, d_2, \ldots, d_n)$: The diagonal matrix of order n with d_i as the i-th diagonal entry, for $1 \leq i \leq n$.
- k-MDS: A matrix B is k-MDS means B^k is an MDS matrix.
- k-NMDS: A matrix B is k-NMDS means B^k is an NMDS matrix.
- $\mathbb{F}_q[L]$: The set of polynomials of L over $\mathbb{F}_q$.

- The matrix $[[1,2,3],[1,3],[2]]$: The 3×3 binary matrix where the nonzero positions are located at the 1st, 2nd, and 3rd positions in the first row; the 1st and 3rd positions in the second row; and the 2nd position in the third row.
- XOR(α): The XOR count of $\alpha \in \mathbb{F}_{2^r}$.
- 0x124: "0x" prefix indicates that the number is written in hexadecimal notation. 0x124 is equivalent to the binary representation 0001 0010 0100.

1 Introduction

Contents

With the increasing adoption of interconnected devices, a significant volume of data is being transmitted through the Internet of Things (IoT). This data often contains sensitive and personal information, necessitating protection against unauthorized access. To achieve various security objectives, such as confidentiality and authenticity, symmetric cryptographic algorithms play a crucial role. These algorithms include block ciphers, stream ciphers, hash functions, and message authentication codes. Among these, block ciphers hold particular significance as fundamental components of symmetric cryptography. They not only provide essential security features but also serve as the foundation for other cryptographic primitives. For example, a secure block cipher can be used as a building block for creating secure hash functions or message authentication codes. To that end, the availability of a *secure block cipher* is one of the fundamental criteria in the design of many cryptographic primitives [1].

A *block cipher* is formally defined as a collection of permutations within a finite message space parametrized by a key from a finite key space. Ideally, each of these permutations should be indistinguishable from a randomly chosen permutation from the entire set of permutations within the message space. Due to the complexity of achieving this ideal security goal, a more practical security notion has emerged. In this

[1] An alternative approach is to utilize cryptographic permutations, such as Keccak [BDPA11] and ASCON [DEMS21], to construct symmetric key cryptographic primitives without depending on block ciphers.

context, a cryptographic primitive is considered secure if no significant vulnerabilities have been discovered over an extended period, typically spanning several years. In the present day, we possess highly efficient and versatile block ciphers of this type. The foremost example is the Advanced Encryption Standard (AES) [DR02], which is the standardized version of the block cipher Rijndael [DR99]. It is widely regarded as the most comprehensively understood construction in this field, and since its publication, no significant vulnerabilities have been identified. Its elegant design, along with its adaptability for numerous applications, positions it as the current state-of-the-art cipher.

Nevertheless, with the expanding range of smaller, more cost-effective connected devices, there may arise scenarios where a cryptographic solution specifically designed to meet highly restrictive demands for performance and efficiency is necessary. Consequently, there is a need for new designs optimized across various lightweight metrics, including hardware cost, power consumption, and latency. Over time, a multitude of these lightweight primitives has emerged. For a comprehensive list, we refer to [BP17].

Most block ciphers are classified as iterated ciphers, where the output is generated by repeatedly applying a fixed key-dependent function r times to the input. This function, referred to as the *round function*, and the block cipher is referred to as a *r-round iterated (block) cipher* [DR02, Section 2.4.1]. The round function typically consists of a simple combination of a nonlinear operation and a linear operation, often referred to as a linear layer. This design paradigm has been extensively studied and includes ciphers like AES [DR02]. However, it is noteworthy that the hash function Cellhash [DGV91] by Daemen et al. introduced the dedicated mixing layers in symmetric key cryptography. The mixing layer in Cellhash needs two XORs for each state bit, and reversing them needs many XORs. It is worth noting that the hash function KECCAK [BDPA11] and the permutation Xoodoo [DHAK18] use mixing layers with a similar feature. The block cipher 3-Way [DGV93] employed a mixing layer inspired by error-correcting codes. Additionally, in many design choices for lightweight cryptographic primitives, established algorithms such as AES are adapted by modifying their components to meet specific lightweight requirements.

1.1 Diffusion Layer

Claude Shannon, in his paper "Communication Theory of Secrecy Systems" [Sha49], introduced the concepts of *confusion* and *diffusion*, which play a significant role in the design of symmetric key cryptographic primitives. The concept of confusion

aims to create a statistical relationship between the ciphertext and message that is too intricate for an attacker to exploit. In general, confusion is achieved through the interaction between nonlinear *Sboxes* and mixing and shuffling processes over multiple rounds. Diffusion means that if we change a single bit of the plaintext, then about half of the bits in the ciphertext should change, and similarly, if we change one bit of the ciphertext, then about half of the plaintext bits should change. This is equivalent to the expectation that encryption schemes exhibit an avalanche effect [WT86]. The purpose of diffusion is to hide the statistical relationship between the ciphertext and the plaintext. In many block ciphers and hash functions, the diffusion property is attained through the use of a linear layer, which can be represented as a matrix. This matrix is designed to produce a significant alteration in the output for a small change in the input.

It is worth noting that the exact meaning of the term diffusion strongly depends on the context in which it is used [DR02]. In this thesis, we will use the term diffusion to refer to the diffusion effect of a linear transformation T, unless explicitly stated otherwise. This effect can be studied by analyzing the pairs $(x, T(x))$. Additionally, in this thesis, we will utilize the term *perfect diffusion* [Vau95], which pertains to the concept of a linear transformation such that changing i components to the vector x results in changing at least $(n - i + 1)$ components to the vector $T(x)$, where n represents the number of components in the vector $T(x)$. The concept of perfect diffusion in cryptography can be formalized in various ways. One approach involves the use of multipermutations, introduced in [SV95, Vau95]. Another method employs *branch numbers*, introduced by Joan Daemen in his doctoral thesis [Dae95], and *Maximum Distance Separable (MDS) matrices* [MS77].

Heys and Tavares [HT94, HT95, HT96] showed that replacing the permutation layer of Substitution Permutation Networks (SPNs) with a diffusive linear transformation can improve the avalanche characteristics of a block cipher, thereby increasing its resistance to differential and linear cryptanalysis. MDS matrices are important components of modern ciphers and hash functions as they offer diffusion properties that enhance security against these types of attacks. Hence, MDS matrices find significant applications in the design of block ciphers and hash functions.

A great deal of research on MDS matrices with cryptography in mind has been done during the period 1994 to 1998. In 1994, Schnorr and Vaudenay [SV95] introduced the concept of multipermutations as a way to formalize diffusion layers. The usefulness of multipermutations in designing cryptographic primitives was demonstrated by Vaudenay [Vau95] in 1995. Between 1994 and 1996, Heys and Tavares

[HT94, HT95, HT96] conducted research on Substitution Permutation Networks (SPNs) and found that replacing the permutation layer with a diffusive linear transformation could improve the avalanche characteristics of the block cipher, making it more resistant to differential and linear cryptanalysis. In 1996, Rijmen et al. were the first to incorporate MDS matrices into the cipher called SHARK [RDP+96]. The utilization of MDS matrices continued in 1997 when Daemen et al. incorporated them into the cipher SQUARE [DKR97]. In 1998, Daemen and Rijmen incorporated a circulant MDS matrix in the design of the widely used cipher AES [DR02]. Schneier et al. then incorporated MDS matrices into the block cipher Twofish [SKW+98, Sch98] between 1998 and 1999. As a result, the effectiveness of MDS matrices in diffusion layers is now widely recognized. The stream cipher MUGI [WFY+02] employs AES MDS matrices in its linear transformations. Additionally, MDS matrices have been utilized in the development of various hash functions, including Whirlpool [BR00c, SS03], SPN-Hash [CYK+12], Maelstrom [FBR06], Grϕstl [GKM+08], and the PHOTON family of lightweight hash functions [GPP11].

The diffusion power of a linear transformation, as defined by a matrix, is measured through its *branch numbers* [Dae95]. While a linear transformation with strong diffusion power, particularly one with an optimal branch number, is crucial for security, it often involves a high hardware implementation cost [2]. To address this challenge, *recursive MDS matrices* have been proposed. A matrix B is defined as a recursive MDS matrix if its exponentiation B^k is an MDS matrix, where k is a positive integer. The implementation of B^k can be achieved by recursively executing the implementation of B, which necessitates k clock cycles. The design of the PHOTON family of hash functions [GPP11] and the LED block cipher [GPPR11] has incorporated recursive MDS matrices based on companion matrices, which can be efficiently implemented using a simple LFSR. Subsequently, researchers have focused on designing recursive MDS matrices, producing a significant number of results [AF15, Ber13, GPV17a, GPV17b, GPV19, SDMS12, TTKS18, WWW13, XTL14].

Therefore, we can categorize the approaches of constructing MDS matrices in two ways: nonrecursive and recursive. The nonrecursive and recursive techniques can be further classified based on whether the matrix is constructed directly or by a search method by enumerating a search space. Nonrecursive direct constructions are mainly obtained from Cauchy and Vandermonde based constructions, while recursive

[2]Note that security can be achieved even without using a linear transformation with a high branch number. However, in such cases, achieving security requires a greater number of rounds compared to the primitives that utilize a linear transformation with a high branch number.

direct constructions use some coding theoretic techniques. For instance, Augot et al. [AF15] employed shortened BCH codes, while Berger [Ber13] utilized Gabidulin codes in their method. Following that, a series of works [GPV17a, GPV17b, GPV19] proposed multiple approaches for the direct construction of recursive MDS matrices from the companion matrices over finite fields. More recently, in [KPSV21], the authors introduced several direct constructions of recursive MDS matrices over finite commutative rings.

While direct construction methods provide the feasibility of obtaining MDS matrices of any order, there is no guarantee of achieving a matrix with the optimal hardware area. This holds true even for smaller sizes. Exhaustive search is currently the only known method that can provide an optimal MDS matrix in terms of area. However, this approach is feasible only when the matrix size is small and the field size is not too large. In the context of nonrecursive approaches, search techniques have been applied to various matrix structures, including *circulant*, *left-circulant*, *Hadamard*, and *Toeplitz matrices*. Significant work has been done in this direction, as demonstrated in [GR15, LS16, PSA+18, SKOP15, SS16, SS17]. On the other hand, many sparse matrix structures, such as *companion*, *Generalized-Feistel-Structure (GFS)* [WWW13], *Diagonal-Serial-Invertible (DSI)* [TTKS18], *sparse DSI* [TTKS18] have been proposed for the construction of recursive MDS matrices.

However, the trade-off between security and efficiency may not be optimal with MDS and recursive MDS matrices. *Near-MDS (NMDS)* matrices have sub-optimal branch numbers, leading to a slower diffusion speed compared to MDS matrices. However, studies such as [ABI+18, BBI+15] have indicated that the use of NMDS matrices, in combination with a well-selected permutation, can enhance security against differential and linear cryptanalysis. Some recent lightweight block ciphers, such as PRIDE [ADK+14], Midori [BBI+15], MANTIS [BJK+16], FIDES [BBK+13] and PRINCE [BCG+12] have utilized NMDS matrices due to their better balance between security and efficiency. As the importance of lightweight symmetric key primitives grows, NMDS matrices are becoming increasingly common in the construction of lightweight block ciphers. However, the study of NMDS matrices has been relatively limited in the literature. In 2017, Li et al. [LW17] studied the construction of NMDS matrices from circulant and Hadamard matrices. In [LW21], the focus is on studying the *recursive NMDS matrices* with the goal of achieving the lowest possible hardware cost. Also, recent studies such as [HYNL21, SYLH22] have presented direct constructions of NMDS codes, which can be utilized to derive nonrecursive NMDS matrices.

1.2 Aims and Contributions

The purpose of this thesis is to expand knowledge on designing diffusion layers for cryptographic primitives. Specifically, we study MDS and Near-MDS matrices and explore their construction in both recursive and nonrecursive settings using various matrix structures. This thesis has five major contributions:

1. *A brief survey and some generalized results on MDS matrices.* This contribution involves a brief survey on MDS matrices. It includes not only a summary of existing results but also the revelation of deep and nontrivial connections among various constructions of MDS matrices. For instance, it reveals that all Vandermonde constructions are equivalent to Cauchy constructions. This contribution includes proof of some folklore results that are used in MDS matrix literature and offers simpler alternative proofs wherever possible. The results of this contribution are given in Chapter 3.

2. *Study of sparse matrices for the construction of recursive MDS and Near-MDS matrices.* This contribution first presents a systematic study of constructing recursive MDS matrices using sparse matrices with low fixed XOR (see Section 2.6.1). It presents new mathematical results and rediscoveries of some existing results on sparse matrices, such as DSI and sparse DSI matrices, and provides some impossibility results. The results of this investigation are presented in Chapter 4. Following this, this contribution introduces a new class of sparse matrices called Diagonal-like sparse (DLS) matrices, where the DSI matrix is a special case of DLS matrix. Then it provides some theoretical results on DLS matrices for the construction of MDS and Near-MDS matrices. To address the impracticality of an exhaustive search for higher order recursive MDS or Near-MDS matrices using DLS matrices, this study introduces some equivalence classes of DLS matrices that help to constrain the search space to a smaller domain. The new class of sparse matrices is thoroughly examined in Chapter 5 and Chapter 6.

3. *Study of constructing Near-MDS matrices from various matrix structures.* The optimal branch number of MDS matrices makes them a preferred choice for designing diffusion layers in many block ciphers and hash functions. However, in lightweight cryptography, Near-MDS matrices with sub-optimal branch numbers offer a better balance between security and efficiency as a diffusion layer, compared to MDS matrices. This contribution delves into the study of Near-MDS matrices,

investigating their construction in both recursive and nonrecursive scenarios. It investigates the theoretical aspects of constructing Near-MDS matrices and evaluates their hardware efficiency. Furthermore, this contribution draws comparisons between the results of Near-MDS and MDS matrices, wherever possible. Chapter 6 thoroughly investigates the construction of Near-MDS matrices from various matrix structures.

4. *Direct constructions of MDS and Near-MDS matrices.* Although an exhaustive search may be suitable for finding small order MDS and Near-MDS matrices, direct constructions are preferred for larger orders, mainly because of the vast search space involved. This contribution introduces new direct constructions of nonrecursive MDS and Near-MDS matrices using generalized Vandermonde matrices. Additionally, it presents some direct constructions of recursive MDS and Near-MDS matrices derived from companion matrices. These constructions are discussed in Chapter 7.

5. *Proposal of a new 64-bit lightweight block cipher with MDS matrix.* While security is the primary concern for cryptographic primitives, efficient implementation in hardware and software is also crucial for lightweight primitives. The high implementation cost of MDS matrices in the round function of lightweight block ciphers poses a challenge. This has led to many lightweight block ciphers avoiding the use of MDS matrices in their design, resulting in the need for a large number of rounds in encryption. To address this challenge, this contribution introduces FUTURE, a new 64-bit lightweight SPN-based block cipher. FUTURE tackles this challenge by carefully selecting a lightweight MDS matrix, which is a composition of four sparse matrices. The specification, design rationale, and security analysis of FUTURE are thoroughly discussed in Chapter 8.

1.3 Outline of this Thesis

This thesis is based on five published papers [GPRS19, GPS19, GPS22a, GPS22b, GPS23a] and one more paper [GPS23b] which are communicated. Chapter 1 presents an overview of the thesis, while Chapter 2 covers the essential preliminary materials needed for the subsequent chapters.

In *Chapter 3*, we provide a brief survey on MDS matrices over finite fields. While most of the results in this chapter are already known, some results and insights are new. We provide a nontrivial and deep interconnection between all the known

Cauchy based constructions and their corresponding Vandermonde based constructions. In [GR15], the authors established the impossibility of involutory or orthogonal *Type-I circulant-like* MDS matrices with even order, but the case of odd orders was left unexplored. In this chapter, we address this gap and present Lemma 3.19 and Lemma 3.21, demonstrating the non-existence of involutory or orthogonal *Type-I circulant-like* MDS matrices with odd order. In Lemma 1 of [LS16], the authors provided a necessary and sufficient condition for the equivalence between two circulant matrices. In Lemma 3.24, we provide a simpler alternative proof for the equivalence.

It also explores interconnections between left-circulant and circulant matrices, as well as between Hankel and Toeplitz matrices. In [LS16] it was proved that left-circulant matrices of order 2^n are not involutory. In Theorem 3.9, we show that this result can easily be derived from the interconnections and known results on circulant matrices. Also, we prove some folklore results that are often used in literature, mostly without formal proofs. The chapter also fills a gap in [AF15, Lemma 1], and provides a corrected version in Lemma 3.26. This chapter is based on the collaborative work [GPRS19] with Kishan Chand Gupta, Sumit Kumar Pandey, and Indranil Ghosh Ray.

The next two chapters discuss the construction of recursive MDS matrices from sparse matrices with low fixed XOR. A brief overview of each chapter is provided below.

In *Chapter 4*, we formalize matrices with low fixed XOR (see Section 2.6.1) and study their properties systematically. Our study starts with a matrix with the minimum number of fixed XOR required, which is one, to construct any recursive MDS matrix. We call such matrices 1-XOR matrices. We provide upper bounds on the number of nonzero elements of 1-XOR matrices of order n when raised to power n. Next, we move on to 2-XOR matrices and provide some impossibility results for matrices of order 5 and 6 for constructing recursive MDS matrices. Finally, in this chapter, we demonstrate the non-existence of an 8-MDS sparse DSI matrix of order 8 over $\mathbb{F}_{2^8}$. This result was previously unsolved in [TTKS18] due to the large search space. However, we are able to drastically reduce the search space by providing some nontrivial theoretical results on sparse DSI matrices. The results of this chapter are derived from the collaborative work [GPS19] with Kishan Chand Gupta and Sumit Kumar Pandey.

In *Chapter 5*, we introduce a new type of sparse matrix called the Diagonal-like sparse (DLS) matrix, which includes the DSI matrix as a specific type. We prove that the value of fixed XOR should be at least $\lceil \frac{n}{2} \rceil$ for an n-MDS DLS matrix of

order n. The exhaustive search using a naive way of finding a higher order recursive MDS matrix using DLS matrices is impractical. In this regard, we present some theoretical results that are used to narrow the search space to a small domain. We also show that an n-MDS DLS matrix with fixed XOR $= \lceil \frac{n}{2} \rceil$ is a permutation similar to some n-MDS sparse DSI matrix over $\mathbb{F}_{2^r}$. Then we generalize the structure of DLS matrices and provide another class of sparse matrices called generalized DLS (GDLS) matrices. Using these matrices, we introduce some lightweight recursive MDS matrices of orders $4, 5, 6$, and 7 that can be implemented with $22, 30, 31$, and 45 XORs over $\mathbb{F}_{2^8}$, respectively. The results match the best known lightweight recursive MDS matrices of orders 4 and 6 and beat the best known matrices of orders 5 and 7. Besides searching over $\mathbb{F}_{2^4}$ and $\mathbb{F}_{2^8}$, we also provide some efficient n-MDS GDLS matrices over $GL(8, \mathbb{F}_2)$ for various orders. The contents of this chapter are based on the collaborative work [GPS22a] with Kishan Chand Gupta and Sumit Kumar Pandey.

In *Chapter 6*, we study the properties of Near-MDS matrices, examining their construction in both recursive and nonrecursive approaches. We provide several theoretical results and explore the hardware efficiency of the construction of Near-MDS matrices. Additionally, we make comparisons between the results of Near-MDS and MDS matrices whenever possible. For the recursive approach, we study the DLS matrices and provide some theoretical results on their use. Some of the results are used to restrict the search space of the DLS matrices. We also show that over a field of characteristic 2, any sparse matrix of order $n \geq 4$ with fixed XOR of 1 cannot be an Near-MDS when raised to a power of $k \leq n$. Following that, we use the GDLS matrices to provide some lightweight recursive Near-MDS matrices of several orders that perform better than the existing matrices in terms of hardware cost or the number of iterations. We examine different structures for the nonrecursive construction of Near-MDS matrices, including circulant and left-circulant matrices, as well as their generalizations such as Toeplitz and Hankel matrices. Proposition 3 in [LW17] demonstrates that circulant matrices of order $n > 4$ cannot be both Near-MDS and involutory over $\mathbb{F}_{2^r}$. In Theorem 6.7, we prove that this result also holds for Toeplitz matrices. Finally, we use GDLS matrices to provide some lightweight Near-MDS matrices that can be computed in one clock cycle. The proposed nonrecursive Near-MDS matrices of orders 4, 5, 6, 7, and 8 can be implemented with 24, 50, 65, 96, and 108 XORs over $\mathbb{F}_{2^4}$, respectively. The contents presented in this chapter are based on the collaborative work [GPS23a] conducted with Kishan Chand Gupta and Sumit Kumar Pandey.

In *Chapter 7*, we present various techniques for direct construction of MDS and Near-MDS matrices over finite fields, in both recursive and nonrecursive approach. In the recursive approach, we begin by establishing a criterion for determining the similarity between a companion matrix and a diagonal matrix. From there, we can represent the companion matrix in terms of a Vandermonde matrix and the diagonal matrix. With the help of determinant expressions for generalized Vandermonde matrices, we present various techniques for constructing recursive MDS and Near-MDS matrices that are derived from the companion matrices. Furthermore, we present direct constructions of nonrecursive MDS and Near-MDS matrices, which are based on two generalized Vandermonde matrices. The results of this chapter are based on the collaborative work [GPS23b] with Kishan Chand Gupta and Sumit Kumar Pandey.

In *Chapter 8*, we present a new 64-bit SPN-based lightweight block cipher, FUTURE, that is designed for minimal latency with low hardware implementation cost. To achieve the best diffusion in the linear layer, FUTURE incorporates an MDS matrix in its round function. The cost of the MDS matrix in FUTURE is optimized by utilizing a specific type of MDS matrix construction. Additionally, by carefully selecting the FUTURE Sbox as a combination of four lightweight Sboxes, we have significantly reduced the implementation cost. Furthermore, FUTURE demonstrates its resistance to various fundamental attacks. This chapter is based on the collaborative work [GPS22b] with Kishan Chand Gupta and Sumit Kumar Pandey.

2

Background and Preliminaries

Contents

2.1 Finite Field

Let $\mathbb{F}_q$ be the finite field containing q elements, where $q = p^r$ for some prime p and a positive integer r. The set of vectors of length n with entries from the finite field $\mathbb{F}_q$ is denoted by $\mathbb{F}_q^n$. The polynomial ring over $\mathbb{F}_q$ in the variable x is denoted as $\mathbb{F}_q[x]$. The algebraic closure of $\mathbb{F}_q$ is denoted as $\bar{\mathbb{F}}_q$, and the multiplicative group is denoted as $\mathbb{F}_q^*$. It is a well established fact that elements of a finite field with characteristic p can be represented as vectors with coefficients in $\mathbb{F}_p$. In other words, there exists a vector space isomorphism from $\mathbb{F}_{p^r}$ to $\mathbb{F}_p^r$ defined by $x = (x_1\alpha_1 + x_2\alpha_2 + \cdots + x_r\alpha_r) \mapsto (x_1, x_2, \ldots, x_r)$, where $\{\alpha_1, \alpha_2, \ldots, \alpha_r\}$ is a basis of $\mathbb{F}_{p^r}$. If α is a primitive element of $\mathbb{F}_{p^r}$, every nonzero element of $\mathbb{F}_{p^r}$ can be expressed as a power of α i.e.,

$\mathbb{F}_{p^r}^* = \{1, \alpha, \alpha^2, \alpha^3, \ldots, \alpha^{p^r-1}\}$. When $q = 2^r$, we use a compact notation to denote the finite field $\mathbb{F}_{2^r}$ by adopting a hexadecimal representation. For instance, $\mathbb{F}_{2^4}$/0x13 represents the finite field $\mathbb{F}_{2^4}$ constructed by the irreducible polynomial $x^4 + x + 1$ over $\mathbb{F}_2$.

The *characteristic of a field* is defined as the smallest positive integer p for which $p\beta = 0$ holds for every nonzero element β in the field. If there is no such integer, the field is said to have characteristic 0. It is also known that if a field has characteristic p, then p must be a prime number. Therefore, for a finite field $\mathbb{F}_q$, its characteristic must be a prime number p. Also, it is a well-known result that if $\mathbb{F}_q$ is a finite field of characteristic p, then $q = p^r$ for some positive integer r.

We denote $M_{k\times n}(\mathbb{F}_q)$ as the set of all matrices with size $k \times n$ over the field $\mathbb{F}_q$. For convenience, we denote the ring of square matrices of order n (matrix of size $n \times n$) over $\mathbb{F}_q$ as $M_n(\mathbb{F}_q)$. The identity matrix of $M_n(\mathbb{F}_q)$ is represented as I_n. The determinant of a matrix A in $M_n(\mathbb{F}_q)$ is denoted by $\det(A)$. A square matrix A is considered nonsingular if its determinant is nonzero, or equivalently, if its rows (or columns) are linearly independent over $\mathbb{F}_q$. The general linear group, consisting of nonsingular $n \times n$ matrices over $\mathbb{F}_q$, is denoted by $GL(n, \mathbb{F}_q)$. We now recall some concepts from coding theory.

2.2 Linear Code

An $[n, k]$ *linear code* $\mathcal{C}$ over the finite field $\mathbb{F}_q$ is a nonempty set $\mathcal{C} \subset \mathbb{F}_q^n$ that forms a k-dimensional linear subspace of $\mathbb{F}_q^n$. The *dual code* of $\mathcal{C}$, denoted as $\mathcal{C}^\perp$, consists of vectors that are orthogonal to all codewords in $\mathcal{C}$:

$$\mathcal{C}^\perp = \{\mathbf{x} \in \mathbb{F}_q^n : \mathbf{x} \cdot \mathbf{c} = \mathbf{0} \text{ for all } \mathbf{c} \in \mathcal{C}\}.$$

As a linear code forms a vector space, all its elements can be expressed in terms of a basis. By knowing the basis of a linear code, we can explicitly describe its codewords. In practice, a *generator matrix* represents a basis of a linear code. Conversely, a *parity check matrix* represents a basis for the dual code. Both generator and parity check matrices play crucial roles in coding theory.

Definition 2.1. *Let $\mathcal{C}$ be an $[n, k]$ linear code over $\mathbb{F}_q$. A generator matrix of $\mathcal{C}$ over $\mathbb{F}_q$ is defined as a $k \times n$ matrix G whose rows form a basis for $\mathcal{C}$. On the other hand, a parity check matrix of $\mathcal{C}$ over $\mathbb{F}_q$ is a $(n-k) \times n$ matrix H such that for every $\mathbf{c} \in \mathbb{F}_q^n$,*

$$\mathbf{c} \in \mathcal{C} \iff H\mathbf{c}^T = \mathbf{0}.$$

In other words, the code $\mathcal{C}$ is the kernel of H in $\mathbb{F}_q^n$.

Remark 2.1. *If an $[n, k]$ linear code $\mathcal{C}$ has a generator matrix G and a parity check matrix H, then the dual code $\mathcal{C}^\perp$ is characterized by having a generator matrix H and a parity check matrix G. Consequently, $\mathcal{C}^\perp$ represents an $n-k$ dimensional linear subspace of $\mathbb{F}_q^n$ over the field $\mathbb{F}_q$.*

As the choice of a basis in a vector space is not unique, a code can have multiple generator matrices that can be transformed into one another by elementary row operations. A generator matrix G is considered to be in standard form if it follows the specific structure $G = [I_k \mid A]$. Here, I_k represents the $k \times k$ identity matrix, and A represents a $k \times (n-k)$ matrix. If we have a generator matrix in the form $G = [I_k \mid A]$, then the corresponding parity check matrix for the linear code $\mathcal{C}$ can be defined as $H = [-A^T \mid I_{n-k}]$.
The *(Hamming) distance* between two vectors $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ and $\mathbf{y} = (y_1, y_2, \ldots, y_n)$, denoted as $d(\mathbf{x}, \mathbf{y})$, is defined as the number of coordinates where they differ. For a linear code $\mathcal{C}$, the *minimum distance* d is defined as the smallest Hamming distance between distinct codewords in $\mathcal{C}$, i.e.,

$$d = \min_{\mathbf{x},\mathbf{y}\in\mathcal{C},\mathbf{x}\neq\mathbf{y}} d(\mathbf{x}, \mathbf{y}) = \min_{\mathbf{x},\mathbf{y}\in\mathcal{C},\mathbf{x}\neq\mathbf{y}} w(\mathbf{x} - \mathbf{y}),$$

where $w(\mathbf{x})$ represents the (Hamming) weight of the vector $\mathbf{x}$ i.e. the number of nonzero components in $\mathbf{x}$.

Since $\mathcal{C}$ is a linear code, an alternative definition for d would be the minimum weight among the nonzero codewords in $\mathcal{C}$. A code with parameters $[n, k, d]$ indicates an $[n, k]$ code with minimum distance d.
The following lemma establishes a connection between the properties of a parity check matrix and the minimum distance d of a linear code $\mathcal{C}$.

Lemma 2.1. *[MS77, Page 33] Let H be a parity check matrix of a code $\mathcal{C}$. Then the code has minimum distance d if and only if the following conditions hold:*

(i) Every set of $d-1$ columns of H is linearly independent.

(ii) There exists a set of d columns of H that are linearly dependent.

Constructing a linear code with large values of $\frac{k}{n}$ and d is desirable in coding theory. However, there is a trade-off between the parameters n, k, and d. For instance, the well-known Singleton bound gives an upper bound on the minimum distance for a code.

Theorem 2.1. *(The Singleton bound)[MS77, Page 33] For an $[n, k, d]$ code $\mathcal{C}$, the minimum distance d of the code satisfies the inequality $d \leq n - k + 1$.*

Definition 2.2. *(MDS code) An $[n, k, d]$ code with $d = n - k + 1$ is referred to as a maximum distance separable code or MDS code for short.*

Remark 2.2. *An $[n, k]$ MDS code is defined as having minimum distance of $n-k+1$. Therefore, it follows that every set of $n - k$ columns of the parity check matrix of an $[n, k]$ MDS code is linearly independent.*

Remark 2.3. *It is known that the dual of an MDS code is also an MDS code [MS77, Page 318]. As a consequence, every set of k columns in the generator matrix of an $[n, k]$ MDS code is linearly independent.*

Theorem 2.2. *[MS77, Page 321] For an $[n, k, d]$ code $\mathcal{C}$ with a generator matrix $G = [I \mid A]$, where A is a $k \times (n - k)$ matrix, the code $\mathcal{C}$ is MDS if and only if every square submatrix formed from any i rows and any i columns, for any $i = 1, 2, \ldots, \min\{k, n - k\}$, of the matrix A is nonsingular.*

Now we briefly record the *MDS conjecture* in the following fact.

Fact 2.1. *(MDS Conjecture)[Hir95][MS77, Page 328] Let C be an $[n, k, d]$ linear MDS code over $\mathbb{F}_q$. Then*

$$n \leq \begin{cases} q + 1, & 2 \leq k \leq q \\ k + 1, & q < k \end{cases}$$

except for $k \in \{3, q - 1\}$ and q is even, in which case it has length at most $q + 2$.

We also like to mention that $[n, 1, n]$, $[n, n - 1, 2]$ and $[n, n, 1]$ are called trivial MDS codes, other MDS codes are called nontrivial.
Now we will briefly discuss another important class of linear code that has many applications in cryptography. In [DL95], the concept of Near-MDS codes is introduced as a relaxation of some constraints of the MDS code. The widely used approach to defining Near-MDS codes is through *generalized Hamming weights* [Wei91].

Definition 2.3. *[Wei91] Let $\mathcal{C}$ be an $[n, k]$ code with $\mathcal{D} \subset \mathcal{C}$ as a subcode of $\mathcal{C}$. The support of $\mathcal{D}$, denoted by $\chi(\mathcal{D})$, is the set of coordinate positions, where not all codewords of $\mathcal{D}$ have zero i.e.*

$$\chi(\mathcal{D}) = \{i : \exists (x_1, x_2, \ldots, x_n) \in \mathcal{D} \text{ and } x_i \neq 0\}.$$

Using the terminology, an $[n, k]$ code is a linear code of rank k and support size at most n. The rank of a vector space is its dimension, and we may use the terms rank and dimension interchangeably.

Example 2.1. *Let $\mathcal{C}$ be the linear code with a generator matrix*

$$G = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}.$$

Then $\chi(\mathcal{C}) = \{1, 2, 3, 5, 6\}$ and $\chi(\mathcal{D}) = \{2, 3, 5, 6\}$ for the subcode $\mathcal{D}$ generated by the second and third rows of G.

Definition 2.4. *[Wei91] For a linear code $\mathcal{C}$, the r-th generalized Hamming weight, denoted as $d_r(\mathcal{C})$, is defined as the cardinality of the minimal support of an r-dimensional subcode of $\mathcal{C}$, where $1 \leq r \leq k$, i.e.*

$$d_r(\mathcal{C}) = \min\{|\chi(\mathcal{D})| : \mathcal{D} \text{ is a subcode of } \mathcal{C} \text{ with rank } r\}.$$

Note that $d_1(\mathcal{C}) = d$ is the minimum distance of $\mathcal{C}$.

Example 2.2. *Consider the linear code $\mathcal{C}$ in Example 2.1. It is easy to check that $d_1(\mathcal{C}) = 2$. By determining the minimal support of all two-dimensional subspaces $\mathcal{D} \subset \mathcal{C}$, we get $d_2(\mathcal{C}) = 4$. Also, there is at least one codeword in $\mathcal{C}$ with a 1 in each position except the fourth position, which implies that $d_3(\mathcal{C}) = 5$.*

Theorem 2.3. *(Monotonicity) [Wei91] For every $[n, k, d]$ linear code, we have*

$$1 \leq d_1(\mathcal{C}) = d < d_2(\mathcal{C}) < d_3(\mathcal{C}) \cdots < d_k(\mathcal{C}) \leq n.$$

Corollary 2.1. *(Generalized Singleton bound) [Wei91] For an $[n, k]$ linear code $\mathcal{C}$, $d_r(\mathcal{C}) \leq n - k + r$. (When $r = 1$, this is the Singleton bound.)*

Theorem 2.4 provides another method to compute the generalized Hamming weight of linear code. Let H be a parity check matrix of $\mathcal{C}$ and let H_i, $1 \leq i \leq n$, be its i-th column vector. Let $< H_i : i \in I >$ be the space generated by the column vectors H_i for $i \in I$.

Theorem 2.4. *[Wei91] For all $r \leq k$,*

$$d_r(\mathcal{C}) = \min\{|I| : |I| - \text{rank}(< H_i : i \in I >) \geq r\}.$$

The following Theorem establishes a connection between the properties of a parity check matrix and the generalized Hamming weight of a linear code $\mathcal{C}$. Although this theorem is well-known, we have not found its proof, so we are providing it below.

Theorem 2.5. *[Wei91, DL95] Let H be a parity check matrix for a linear code $\mathcal{C}$. Then $d_r(\mathcal{C}) = \delta$ if and only if the following conditions hold:*

(i) Any $\delta - 1$ columns of H have rank greater or equal to $\delta - r$.

(ii) There exist δ columns in H of rank $\delta - r$.

Proof. For any $I \subset \{1, 2, \ldots, n\}$, let $S(I) =< H_i : i \in I >$ be the space spanned by the vectors H_i for $i \in I$, where H_i denotes the i-th column of the parity check matrix H of $\mathcal{C}$. Let

$$S^{\perp}(I) = \left\{ \mathbf{x} \in \mathcal{C} : x_i = 0 \text{ for } i \notin I \text{ and } \sum_{i \in I} x_i H_i = 0 \right\}.$$

Then $\text{rank}(S(I))+\text{rank}(S^{\perp}(I)) = |I|$.

Let $d_r(\mathcal{C}) = \delta$, and we will prove that both conditions hold. To do so, let us assume for the sake of contradiction that there exist some $\delta - 1$ columns of H, say $H_{i_1}, H_{i_2}, \ldots, H_{i_{\delta-1}}$, with rank $\leq \delta - r - 1$.

Now let $I = \{i_1, i_2, \ldots, i_{\delta-1}\} \subset \{1, 2, \ldots, n\}$. Then $\text{rank}(S(I)) \leq \delta - r - 1$. Thus, we have

$$\begin{aligned} \text{rank}(S^{\perp}(I)) &= |I| - \text{rank}(S(I)) \\ &\geq \delta - 1 - (\delta - r - 1) = r. \end{aligned}$$

Therefore, we have $\text{rank}(S^{\perp}(I)) \geq r$. Also, by the construction, $S^{\perp}(I)$ is a subcode of $\mathcal{C}$ and $|\chi(S^{\perp}(I))| \leq \delta - 1$. This leads to a contradiction since $d_r(\mathcal{C}) = \delta$. Therefore, we can conclude that any $\delta - 1$ columns of H have rank greater or equal to $\delta - r$.

Since $d_r(\mathcal{C}) = \delta$, there exist a subcode $\mathcal{D}$ of $\mathcal{C}$ with $\text{rank}(\mathcal{D}) = r$ and $|\chi(\mathcal{D})| = d_r(\delta)$. Let $I = \chi(\mathcal{D})$. Now we will show that $\mathcal{D} = S^{\perp}(I)$.

Let $\mathbf{c} = (c_1, c_2, \ldots, c_n) \in \mathcal{D}$ be a codeword. Then we have

$$\begin{aligned}
&\sum_{i=1}^{n} c_i H_i = \mathbf{0} \\
\implies &\sum_{i\in I} c_i H_i + \sum_{i\notin I} c_i H_i = \mathbf{0} \\
\implies &\sum_{i\in I} c_i H_i = \mathbf{0} \quad [\text{Since } c_i = 0 \ \forall i \notin I = \chi(\mathcal{D})] \\
\implies &\mathbf{c} \in S^{\perp}(I) \\
\implies &\mathcal{D} \subset S^{\perp}(I).
\end{aligned}$$

If possible, let $\text{rank}(S^{\perp}(I)) = r' > r$. Now since $\text{rank}(S(I))+\text{rank}(S^{\perp}(I)) = |I|$, we have

$$\begin{aligned}
&|I| - \text{rank}(S(I)) = r' > r \\
\implies &d_{r'}(\mathcal{C}) \leq |I| = \delta \quad [\text{By Theorem 2.4}].
\end{aligned}$$

But by the monotonicity of generalized Hamming weights we must have

$$\delta = d_r(\mathcal{C}) < d_{r'}(\mathcal{C}) \leq \delta,$$

which is a contradiction. Hence, we must have $\text{rank}(\mathcal{D}) = \text{rank}(S^{\perp}(I))$ and $\mathcal{D} = S^{\perp}(I)$. Thus,

$$\text{rank}(S(I)) = |I| - r = \delta - r.$$

Therefore, there exist δ columns in H of rank $\delta - r$.

For the converse part, assume that both the conditions hold. From Condition (ii), we know that there exist some $I \subset \{1, 2, \ldots, n\}$ with $|I| = \delta$ such that $\text{rank}(S(I)) = \delta - r$. This implies that

$$\text{rank}(S^{\perp}(I)) = |I| - \text{rank}(S(I)) = r.$$

Since $|I| - \text{rank}(S(I)) = r$, by Theorem 2.4, we have $d_r(\mathcal{C}) \leq \delta$.

If possible, let $d_r(\mathcal{C}) = \delta - t$ for some $t \geq 1$. Now by Theorem 2.4, there exist some $I' \subset \{1, 2, \ldots, n\}$ with $|I'| = \delta - t$ such that

$$\begin{aligned}
&|I'| - \text{rank}(S(I)) \geq r \\
\implies &\text{rank}(S(I)) \leq |I'| - r \\
\implies &\text{rank}(S(I)) \leq \delta - t - r.
\end{aligned}$$

Therefore, there exist $|I'| = \delta - t$ many columns, say $H_{i_1}, H_{i_2}, \ldots, H_{i_{\delta-t}}$, of H of

rank $\leq \delta - t - r$. Now by adding any other $t-1$ columns of H to that $\delta - t$ columns we have $\delta - 1$ columns, say $H_{i_1}, H_{i_2}, \ldots, H_{i_{\delta-t}}, H_{i_{\delta-t+1}}, \ldots, H_{i_{\delta-1}}$, of H of rank $\leq (\delta - t - r) + (t-1) = \delta - r - 1 < \delta - r$. This leads to a contradiction to condition (i). Hence, we must have $d_r(\mathcal{C}) = \delta$. □

Definition 2.5. *(NMDS code)[DL95] A linear $[n, k]$ code $\mathcal{C}$ is said to be Near-MDS or NMDS if*

$$d_1(\mathcal{C}) = n - k \quad \text{and} \quad d_i(\mathcal{C}) = n - k + i, \quad \text{for } i = 2, 3, \ldots, k.$$

Remark 2.4. *From the monotonicity of generalized Hamming weights, we can say that an $[n, k]$ linear code is NMDS if and only if $d_1(\mathcal{C}) = n-k$ and $d_2(\mathcal{C}) = n-k+2$.*

Theorem 2.5 provides the following useful result on the NMDS code.

Lemma 2.2. *[DL95] For an $[n, k]$ code $\mathcal{C}$ with a parity check matrix H, the code $\mathcal{C}$ is NMDS if and only if the matrix H satisfies the following conditions:*

(i) Every set of $n-k-1$ columns of H is linearly independent.

(ii) There exists a set of $n-k$ columns of H that are linearly dependent.

(iii) Any set of $n-k+1$ columns of H is of full rank.

Proof. Let $\mathcal{C}$ be an NMDS code. Therefore, we have $d_1 = n-k$ and $d_2 = n-k+2$. Since d_1 is the minimum distance of $\mathcal{C}$, from Lemma 2.1, we can say that $d_1 = n-k$ if and only if any $n-k-1$ columns of H are linearly independent and there exist some $n-k$ columns that are linearly dependent. Moreover, Theorem 2.5 implies that $d_2 = n-k+2$ if and only if any $n-k+1$ columns of H have rank greater or equal to $(n-k+2)-2 = n-k$ and there exist $n-k+2$ columns of H of rank $(n-k+2)-2 = n-k$. Since H is a parity check matrix of $\mathcal{C}$, we have $\text{rank}(H) = n-k$. Therefore, we can conclude that $d_2 = n-k+2$ if and only if any $n-k+1$ columns of H are of full rank. Hence, the lemma. □

It can be deduced from the properties of the generalized Hamming weights that the dual of an NMDS code is also an NMDS code.

Lemma 2.3. *[DL95] If a linear $[n, k]$ code is NMDS, then its dual code is also NMDS.*

Corollary 2.2. *[DL95] A linear $[n, k]$ code $\mathcal{C}$ is NMDS if and only if $d(\mathcal{C}) + d(\mathcal{C}^\perp) = n$, where $d(\mathcal{C})$ and $d(\mathcal{C}^\perp)$ denote the minimum distance of the code $\mathcal{C}$ and its dual $\mathcal{C}^\perp$, respectively.*

One can infer from Lemma 2.3 that a generator matrix of a linear $[n, k]$ NMDS code must satisfy conditions similar to those in Lemma 2.2.

Lemma 2.4. *[DL95] For an $[n, k]$ code $\mathcal{C}$ with a generator matrix G, the code $\mathcal{C}$ is NMDS if and only if the matrix G satisfies the following conditions:*

(i) Every set of $k-1$ columns of G is linearly independent.

(ii) There exists a set of k columns of G that are linearly dependent.

(iii) Any set of $k+1$ columns of G is of full rank.

Remark 2.5. *It is worth noting that not all $[n, k, n-k]$ codes are necessarily NMDS codes. For example, consider the linear code $\mathcal{C}$ with generator matrix*

$$G = \begin{bmatrix} 1 & 0 & 0 & \alpha^2 & \alpha & 0 \\ 0 & 1 & 0 & \alpha & \alpha & 0 \\ 0 & 0 & 1 & \alpha & 0 & \alpha \end{bmatrix}$$

over the finite field $\mathbb{F}_{2^2}$, where α is a root of the constructing polynomial $x^2 + x + 1$. Then it can be checked that $\mathcal{C}$ is an $[6, 3, 3]$ code. Also, by determining the minimal support of all two-dimensional subspaces $\mathcal{D} \subset \mathcal{C}$, we get $d_2(\mathcal{C}) = 4 < 5$. This value is achieved by the subspace spanned by the first two rows of the generator matrix G. Hence, $\mathcal{C}$ is not an NMDS code.

Almost-MDS codes, introduced in [DB96], are closely related to NMDS codes.

Definition 2.6. *(AMDS code)[DB96] An $[n, k, d]$ code $\mathcal{C}$ is said to be Almost-MDS or AMDS code if $d = n - k$.*

As pointed out in Remark 2.5, not every AMDS code is NMDS, but for large n both notions coincide.

Theorem 2.6. *[DL95] If $n > k + q$, every $[n, k, n-k]$ code over $\mathbb{F}_q$ is NMDS.*

From Corollary 2.2, we have the following fact, which serves as an alternative definition of an NMDS code.

Fact 2.2. *A linear $[n, k]$ code $\mathcal{C}$ is NMDS if and only if both the code $\mathcal{C}$ and its dual $\mathcal{C}^\perp$ are AMDS codes.*

We close this section by presenting Lemma 2.5, which will be useful in this thesis. To prove this lemma, we need the following result from linear algebra.

Theorem 2.7. *[RB00, Theorem 3.5.4] Let A be a $k \times k$ matrix and B be a $k \times l$ matrix. If A is nonsingular, then the rank of AB is equal to the rank of B.*

Lemma 2.5. *Let A be a $k \times k$ nonsingular matrix and G be a generator matrix of an $[n, k]$ linear code $\mathcal{C}$. Then AG is also a generator matrix of the code $\mathcal{C}$.*

Proof. We know that the rows of the generator matrix G form a basis for the linear code $\mathcal{C}$ and $\text{rank}(G) = k$. Also, since A is nonsingular, according to Theorem 2.7, we have $\text{rank}(AG) = \text{rank}(G) = k$. Therefore, all k rows of AG are linearly independent.

Note that each row of AG is a linear combination of the rows of G. Therefore, each row of AG represents a codeword of $\mathcal{C}$, and these rows are linearly independent. Consequently, the rows of AG form a basis for $\mathcal{C}$. Therefore, AG is also a generator matrix of the code $\mathcal{C}$. □

2.3 MDS and Near-MDS matrices

We will now explore *MDS* and *NMDS* matrices, which have notable cryptographic applications. The concept of MDS and NMDS matrices is derived from the MDS and NMDS codes, respectively.

Remark 2.6. *Generally, the matrix A in the generator matrix $G = [I \mid A]$ of an $[n, k]$ code $\mathcal{C}$ is considered an MDS or NMDS matrix depending on whether the code $\mathcal{C}$ is MDS or NMDS. Since square matrices are typically used in practice, for the sake of simplicity, we will consider the $[2n, n]$ code instead of the generic form of the $[n, k]$ code throughout the rest of this thesis.*

Definition 2.7. *[RDP+96] Consider a finite field $\mathbb{F}_q$ and an integer n. Let $x \to A \times x$ be a mapping from $\mathbb{F}_q^n$ to $\mathbb{F}_q^n$, where A is an $n \times n$ matrix. We define A as an MDS matrix if the set of all pairs $(x, A \times x)$ forms an MDS code i.e. a linear code of dimension n, length $2n$ and minimum distance $n + 1$.*

Therefore, from Theorem 2.2, we have another characterization of an MDS matrix.

Fact 2.3. *A square matrix A is considered an MDS matrix if and only if all of its square submatrices are nonsingular.*

The diffusion power of a linear transformation, as specified by a matrix, is quantified by its *branch numbers* [Dae95].

Definition 2.8. *[Dae95] The differential branch number, $\beta_d(A)$, of a matrix A of order n over the finite field $\mathbb{F}_{2^r}$ is defined as the smallest number of nonzero components in both the input vector x and the output vector Ax, as we consider all nonzero x in $\mathbb{F}_{2^r}^n$ i.e.*

$$\beta_d(A) = \min_{x \neq \mathbf{0}}(w(x) + w(Ax)),$$

where $w(x)$ represents the number of nonzero components in the vector x.

Definition 2.9. *[Dae95] The linear branch number, $\beta_l(A)$, of a matrix A of order n over the finite field $\mathbb{F}_{2^r}$ is defined as the smallest number of nonzero components in both the input vector x and the output vector $A^T x$, as we consider all nonzero x in $\mathbb{F}_{2^r}^n$ i.e.*

$$\beta_l(A) = \min_{x \neq \mathbf{0}}(w(x) + w(A^T x)),$$

where $w(x)$ represents the number of nonzero components in the vector x.

Remark 2.7. *[DR02, Page 144] The differential branch number $\beta_d(A)$ of a matrix A is equal to the minimum distance of the linear code $\mathcal{C}$ generated by the matrix $[I \mid A]$. Furthermore, $\beta_l(A)$ is equivalent to the minimum distance of the dual code $\mathcal{C}^\perp$ of $\mathcal{C}$.*

Remark 2.8. *[DR02, Page 132] It is important to note that the maximum value for both $\beta_d(A)$ and $\beta_l(A)$ is $n+1$. While $\beta_d(A)$ and $\beta_l(A)$ are not always equal, a matrix with the highest possible differential or linear branch number will have the same value for both.*

Therefore, the following fact is another characterization of MDS matrix.

Fact 2.4. *A square matrix A of order n is MDS if $\beta_d(A) = \beta_l(A) = n + 1$.*

The goal of lightweight cryptography is to design ciphers that can be implemented efficiently, in addition to ensuring security. Efficiency is assessed through various parameters, including the gate complexity of its hardware implementation, time taken (measured in clock cycles), and power consumption. The cost of implementation, in terms of space, is directly correlated with gate complexity, while throughput is linked to the time taken. Achieving MDS matrices that demonstrate efficiency across all parameters is nearly impossible, leading to a trade-off between different factors in the quest for suitable matrices. When space is not a limitation, the focus is on matrices that demand fewer clock cycles and consume less energy. In contrast, when space is

a constraint, the search is directed towards matrices that require fewer gates. One proposed method for reducing gate complexity is the use of *recursive MDS matrices*.

Definition 2.10. *Consider a positive integer k. A matrix B is defined as a recursive MDS or k-MDS matrix if the matrix $A = B^k$ is MDS. If B is k-MDS, we can say that B yields an MDS matrix.*

Example 2.3. *For example, the matrix*

$$B = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & \alpha & 0 & 0 \end{bmatrix}$$

is 22-MDS, where α is a primitive element of the field $\mathbb{F}_{2^4}$ with $\alpha^4 + \alpha + 1 = 0$.

We will now discuss NMDS matrices, which have numerous uses in lightweight cryptographic primitives. The concept originates from coding theory, specifically from the NMDS codes.

Definition 2.11. *A matrix A of order n is said to be an NMDS matrix if $[I \mid A]$ is a generator matrix of some $[2n, n]$ linear NMDS code.*

By Remark 2.7, we know that the differential branch number $\beta_d(A)$ of a matrix A is equal to the minimum distance of the linear code $\mathcal{C}$ with the generator matrix $[I \mid A]$. Likewise, $\beta_l(A)$ is equivalent to the minimum distance of $\mathcal{C}^\perp$. As a result, we can characterize an NMDS matrix as follows.

Fact 2.5. *[LW17] A matrix A of order n is called a NMDS matrix if $\beta_d(A) = \beta_l(A) = n$.*

The following lemma is another way to characterize an NMDS matrix.

Lemma 2.6. *[LW17, VR06] For $n \geq 2$, a non-MDS matrix A of order n is classified as NMDS if and only if, for every $1 \leq g \leq n-1$, all $g \times (g+1)$ and $(g+1) \times g$ submatrices of A contain at least one nonsingular $g \times g$ submatrix.*

In Lemma 2.6, if we assume $g = 1$, we can deduce that there is at most one zero in each row and each column of a NMDS matrix. Hence, we have the corollary as follows.

Corollary 2.3. *A NMDS matrix A of order n must contain at least $n^2 - n$ nonzero elements.*

Definition 2.12. *Consider a positive integer k. A matrix B is defined as a recursive NMDS or k-NMDS matrix if the matrix $A = B^k$ is NMDS. If B is k-NMDS, we can say that B yields an NMDS matrix.*

Example 2.4. *The matrix in Example 2.3 is a recursive NMDS matrix with $k = 10$.*

MDS matrices are widely employed in modern block ciphers to enhance the diffusion property. Typically, separate modules are utilized for encryption and decryption operations. However, in [YTH96], Youssef et al. introduced a specific class of SPNs, which utilizes the same network for both encryption and decryption. The innovative approach involved the utilization of involutory MDS matrices to incorporate diffusion.

Definition 2.13. *An involutory matrix is defined as a square matrix A that satisfies the condition $A^2 = I$, or equivalently, $A = A^{-1}$.*

In the context of lightweight cryptographic primitives, employing an involutory or orthogonal matrix permits both encryption and decryption operations using identical or nearly identical circuitry, resulting in an equivalent implementation cost for both processes.

Definition 2.14. *An orthogonal matrix is defined as a square matrix A that satisfies the condition $AA^T = I$.*

In Chapter 3 and Chapter 6, we will delve into the detailed discussion of the involutory and orthogonal properties of MDS and NMDS matrices constructed from various matrix structures.

2.4 Structural Properties of MDS and NMDS matrices

Definition 2.15. *A matrix D of order n is said to be diagonal if $(D)_{i,j} = 0$ for $i \neq j$.*

Using the notation $d_i = (D)_{i,i}$, the diagonal matrix D can be represented as $diag(d_1, d_2, \ldots, d_n)$. It is evident that the determinant of D is given by $\det(D) = \prod_{i=1}^{n} d_i$. Therefore, the diagonal matrix D is nonsingular if and only if $d_i \neq 0$ for $1 \leq i \leq n$.

The multiplication of a row of a matrix by a nonzero scalar is one of the elementary row operations. It is worth noting that both the MDS and NMDS properties are preserved under these operations. Consequently, we can establish the following lemmas.

Lemma 2.7. *If A is an MDS matrix over $\mathbb{F}_{2^r}$, then A', obtained by multiplying a row (or column) of A by any element $c \in \mathbb{F}_{2^r}^*$, will also be an MDS matrix.*

Proof. Take an arbitrary square submatrix B' of A'. Suppose B is the corresponding submatrix of A. If the submatrix contains the row (column) in which c has multiplied, then $\det(B') = c \cdot \det(B)$ otherwise $\det(B') = \det(B)$. Since A is an MDS matrix, we have $\det(B') \neq 0$. Therefore, A' is MDS. □

Lemma 2.8. *If A is an NMDS matrix over $\mathbb{F}_{2^r}$, then A', obtained by multiplying a row (or column) of A by any element $c \in \mathbb{F}_{2^r}^*$, will also be an NMDS matrix.*

Proof. Take B' be an arbitrary $g \times (g+1)$ (or $(g+1) \times g$) submatrix of A'. Suppose B is the corresponding submatrix of A. Since A is an NMDS matrix, B must have a nonsingular $g \times g$ submatrix I. Let I' be the corresponding submatrix of B'. If the submatrix I' contains the row (or column) in which c has multiplied, then $\det(I') = c \cdot \det(I)$ otherwise $\det(I') = \det(I)$. Thus, B' contains a nonsingular $g \times g$ submatrix I'. Therefore, by Lemma 2.6, A' is also a NMDS matrix. □

Let $D = diag(c_1, c_2, \ldots, c_n)$ be a diagonal matrix. Then by the multiplication DA (or AD) it means multiply the i-th row (or i-th column) of A by c_i for $1 \leq i \leq n$. Hence, we can generalize the Lemma 2.7 and Lemma 2.8 as follows.

Corollary 2.4. *Let A be an MDS (NMDS) matrix, then for any nonsingular diagonal matrices D_1 and D_2, D_1AD_2 will also be an MDS (NMDS) matrix.*

It is worth noting that the converse of Corollary 2.4 holds true as well.

Corollary 2.5. *Let B be a recursive MDS (NMDS) matrix, then for any nonsingular diagonal matrix D, DBD^{-1} will also be a recursive MDS (NMDS) matrix.*

Proof. Suppose D is a nonsingular diagonal matrix and B is k-MDS (k-NMDS) i.e. B^k is an MDS (NMDS) matrix. Then we have

$$(DBD^{-1})^k = \underbrace{DBD^{-1} \cdot DBD^{-1} \cdot \ \ldots \ \cdot DBD^{-1}}_{k\text{-times}} = DB^qD^{-1}$$

Now since D is a nonsingular diagonal matrix and B^k is an MDS (NMDS) matrix, from Corollary 2.4, we can say that DB^kD^{-1} is again an MDS (NMDS) matrix. Hence, DBD^{-1} is a recursive MDS (NMDS) matrix. More specifically DBD^{-1} is k-MDS (k-NMDS). □

In the following corollary, we mention an important property of MDS and NMDS matrices.

Corollary 2.6. *If A is an MDS (NMDS) matrix, then its transpose A^T is also an MDS (NMDS) matrix.*

Proof. Consider an arbitrary submatrix of order k from A^T by choosing say $i_1, i_2, i_3, \ldots, i_k$-th rows and $j_1, j_2, j_3, \ldots, j_k$-th columns. Denote this submatrix as $A^T(i_1, i_2, \ldots, i_k|j_1, j_2, \ldots, j_k)$.

It is easy to check that

$$A^T(i_1, i_2, \ldots, i_k|j_1, j_2, \ldots, j_k) = A(j_1, j_2, \ldots, j_k|i_1, i_2, \ldots, i_k)^T.$$

Now

$$\begin{aligned}\det(A^T(i_1, i_2, \ldots, i_k|j_1, j_2, \ldots, j_k)) &= \det(A(j_1, j_2, \ldots, j_k|i_1, i_2, \ldots, i_k)^T) \\ &= \det(A(j_1, j_2, \ldots, j_k|i_1, i_2, \ldots, i_k)).\end{aligned}$$

Therefore, $\det(A(j_1, j_2, \ldots, j_k|i_1, i_2, \ldots, i_k)) \neq 0 \implies \det(A^T(i_1, i_2, \ldots, i_k|j_1, j_2, \ldots, j_k)) \neq 0$. Hence, the result. □

Since each square submatrix of an MDS matrix is nonsingular, we have the following result for MDS matrices.

Fact 2.6. *Every square submatrix of an MDS matrix is also an MDS matrix.*

Remark 2.9. *It should be noted that a square submatrix of an NMDS matrix may not be an NMDS matrix. For example, consider the matrix*

$$A = \begin{bmatrix} 0 & \alpha & 1 & \alpha+1 \\ \alpha+1 & 0 & \alpha & 1 \\ 1 & \alpha+1 & 0 & \alpha \\ \alpha & 1 & \alpha+1 & 0 \end{bmatrix}$$

over $\mathbb{F}_{2^4}$, where α is a root of the constructing polynomial $x^4 + x + 1$. Then it can be checked that A is an NMDS matrix. However, the 2×2 submatrix $\begin{bmatrix} 1 & \alpha+1 \\ \alpha & 1 \end{bmatrix}$ of A is an MDS matrix.

Corollary 2.7. *The inverse of an MDS matrix is MDS.*

Proof. Suppose $G = [I \mid A]$ is a generator matrix of an MDS code. Elementary row operation change $G = [I \mid A]$ to $G' = [A^{-1} \mid I]$. As elementary row operations do not change the code, G' is also generator matrix of the MDS code. Consequently, the code defined by the matrix $[I \mid A^{-1}]$ possesses the same minimum distance. Hence, it can be concluded that A^{-1} is also an MDS matrix. □

Remark 2.10. *Note that an NMDS matrix can be singular. For example, the NMDS matrix A in Remark 2.9 is singular.*

If the matrix A is nonsingular, we can apply the proof of Corollary 2.7 to NMDS matrices. Therefore, we obtain the following result regarding NMDS matrices.

Lemma 2.9. *For a nonsingular NMDS matrix A, its inverse A^{-1} is also an NMDS matrix.*

Definition 2.16. *Let ρ be an element of the symmetric group $\mathcal{S}_n$ (set of all permutations over the set $\{1, 2, \ldots, n\}$). Then by $\rho = [i_1, i_2, i_3, \ldots, i_n]$, where $1 \leq i_j \leq n$ for $j = 1, 2, 3, \ldots, n$, we mean $\rho = \begin{pmatrix} 1 & 2 & 3 & \ldots & n \\ i_1 & i_2 & i_3 & \ldots & i_n \end{pmatrix}$ i.e. $1 \to i_1$, $2 \to i_2$, $\ldots$, $n \to i_n$.*

Then the product of two permutations $\rho_1 = [i_1, i_2, i_3, \ldots, i_n]$ and $\rho_2 = [j_1, j_2, j_3, \ldots, j_n]$ is given by $\rho_1 \cdot \rho_2 = [i_{j_1}, i_{j_2}, i_{j_3}, \ldots, i_{j_n}]$ and the inverse of a permutation $\rho = [i_1, i_2, i_3, \ldots, i_n]$ is the permutation $\delta = [j_1, j_2, j_3, \ldots, j_n]$ such that $\rho \cdot \delta = \delta \cdot \rho = [1, 2, 3, \ldots, n]$.

Example 2.5. *For the two permutations $\rho_1 = [2, 3, 4, 5, 1, 6]$ and $\rho_2 = [1, 4, 3, 2, 6, 5]$ over $\mathcal{S}_6$, their product is given by*

$$\rho_1 \cdot \rho_2 = [2, 5, 4, 3, 6, 1] \text{ and } \rho_2 \cdot \rho_1 = [4, 3, 2, 6, 1, 5].$$

The inverse of the permutation $\rho_1 = [2, 3, 4, 5, 1, 6]$ is given by $\delta = [5, 1, 2, 3, 4, 6]$.

Definition 2.17. *[LS16] An index permutation σ of an ordered set $\{c_1, c_2, \ldots, c_n\}$ is a permutation that rearranges the indices of the elements.*

Consider an index permutation σ on the ordered set $\{c_1, c_2, c_3, c_4, c_5\}$, where $\sigma(i) = 6 - i$. Then the resulting ordered set with respect to σ will be $\{c_5, c_4, c_3, c_2, c_1\}$.

Definition 2.18. *A permutation matrix P of order n related to a permutation $\rho = [i_1, \quad i_2, \quad i_3, \quad \ldots, \quad i_n]$ is the binary matrix obtained by rearranging the rows (or columns) of the identity matrix of order n according to the permutation ρ.*

In this thesis, we will use the row permuted identity matrix to represent permutation matrices. For instance, the permutation matrix P related to the permutation $[4, 2, 3, 1]$ is given by

$$P = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}.$$

Note that a permutation matrix is invertible and the inverse of P is the transpose of P, i.e. $P^{-1} = P^T$. The product of two permutation matrices results in a permutation matrix.

Definition 2.19. *Let ρ be an element of the symmetric group S_n. Then ρ is called a k length cycle or k-cycle, written $(j_1\ j_2\ j_3\ \ldots\ j_k)$, if $\rho = \begin{pmatrix} j_1 & j_2 & j_3 & \cdots & j_k \\ j_2 & j_3 & j_4 & \cdots & j_1 \end{pmatrix}$ i.e. $j_1 \to j_2$, $j_2 \to j_3$, $\ldots$, $j_k \to j_1$.*

For example, the permutation $\rho = [3, 2, 4, 1, 5]$ can be written as $(1\ 3\ 4)$. So ρ is a 3-cycle in S_5.
Now, we will discuss the following result which will be beneficial in the subsequent chapters.

Lemma 2.10. *For any permutation matrix P related to some permutation ρ and any diagonal matrix D, we have $DP = PD_1$ for some diagonal matrix D_1.*

Proof. Let P be a permutation matrix related to the permutation $\rho = [i_1, \quad i_2, \quad i_3, \quad \ldots\ , i_n]$ and $D = diag(d_1, d_2, \ldots, d_n)$. By the multiplication PD, the rows of D are permuted according to the permutation ρ. Similarly, with DP, the columns of D are permuted.

Since P is obtained from the identity matrix of order n by permuting the rows according to ρ, the j-th column of P is the i_j-th column of the identity matrix for $j = 1, 2, \ldots, n$. Thus, we have $(DP)_{i_j,j} = \ d_{i_j}$ for $j = 1, 2, \ldots, n$ and $(DP)_{i,j} = 0$ for others i, j. Also, for $j = 1, 2, \ldots, n$, we have $(PD)_{i_j,j} = \ d_j$ and $(PD)_{i,j} = 0$ for others i, j. Therefore, we have $DP = PD_1$ where $D_1 = diag(d_{i_1}, d_{i_2}, \ldots, d_{i_n})$. □

Permuting rows (or columns) does not change the branch number of a matrix. So we have the following result from [LS16] with a different proof. In this thesis, our

focus is mainly on MDS and NMDS matrices, which have equal differential and linear branch numbers. For convenience, we will refer $\beta_d(A)$ and $\beta_l(A)$ simply as the branch number, denoted as β_A.

Lemma 2.11. *The branch numbers of the matrices A and PAQ are identical for any permutation matrices P and Q.*

Proof. Suppose that x is the nonzero vector such that

$$w(x) + w(Ax) = \beta_A.$$

Note that the inverse of a permutation matrix and product of two permutation matrices is again a permutation matrix. Also, multiplication with a permutation matrix does not change the weight of a vector. Therefore, for $y = Q^{-1}x$ we have

$$\begin{aligned} w(y) + w(PAQy) &= w(Q^{-1}x) + w(PAx) \\ &= w(x) + w(Ax) = \beta_A. \end{aligned}$$

Since $\beta_{PAQ} = \min_{y \neq \mathbf{0}} (w(y) + w(PAQy))$, we have $\beta_{PAQ} \leq \beta_A$.

Again, suppose that x is the nonzero vector such that $w(x) + w(PAQx) = \beta_{PAQ}$. Let $y = Qx$. Now

$$\begin{aligned} w(y) + w(Ay) &= w(Qx) + w(AQx) \\ &= w(x) + w(PAQx) = \beta_{PAQ}. \end{aligned}$$

Therefore, $\beta_A \leq \beta_{PAQ}$. Hence, $\beta_A = \beta_{PAQ}$. □

Definition 2.20. *Two matrices A and B are said to be permutation equivalent, denoted by $A \sim B$, if there exist two permutation matrices P and Q such that $B = PAQ$.*

It can be verified that the relation $\sim$ satisfies the properties of an equivalence relation. Also, based on Fact 2.11, it is evident that permutation equivalent matrices possess identical branch numbers. Consequently, we can derive the following result concerning MDS and NMDS matrices.

Corollary 2.8. *If A is an MDS (NMDS) matrix, then for any permutation matrices P and Q, PAQ is an MDS (NMDS) matrix.*

Corollary 2.9. *Let B be a recursive MDS (NMDS) matrix, then for any permutation matrix P, PBP^{-1} will also be a recursive MDS (NMDS) matrix.*

Definition 2.21. *We will call a matrix A_1 to be diagonal (permutation) similar to a matrix A_2 if $A_1 = DA_2D^{-1}$ ($A_1 = PA_2P^{-1}$) for some nonsingular diagonal matrix D (permutation matrix P).*

Fact 2.7. *Diagonal (permutation) similar to a k-MDS (k-NMDS) matrix is again a k-MDS (k-NMDS) matrix.*

2.5 Various Matrix Structures for the Construction of MDS and NMDS Matrices

Various techniques have been introduced for designing MDS and NMDS matrices, which can be primarily classified into nonrecursive and recursive approaches. In the nonrecursive approach, the matrix itself is MDS or NMDS, whereas in the recursive approach, a sparse matrix B is considered such that the k-th power of B, where k is a positive integer, results in an MDS or NMDS matrix. We will now discuss some matrix structures that are utilized in constructing an MDS or NMDS matrix in nonrecursive setting.

2.5.1 Matrix structures for nonrecursive approaches

Definition 2.22. *(Cauchy matrix) Given* $\{x_1, x_2, ..., x_n\} \subseteq \mathbb{F}_{2^r}$ *and* $\{y_1, y_2, ..., y_n\} \subseteq \mathbb{F}_{2^r}$ *such that* $x_i + y_j \neq 0$ *for all* $1 \leq i, j \leq n$*, then the matrix* $A = (a_{i,j})$, $1 \leq i, j \leq n$*, where* $a_{i,j} = \frac{1}{x_i + y_j}$ *is called a Cauchy matrix.*

It is known that [MS77, Page 323]

$$\det(A) = \frac{\prod_{1 \leq i < j \leq n} (x_j - x_i)(y_j - y_i)}{\prod_{1 \leq i,j \leq n} (x_i + y_j)}.$$

If the values x_i and y_j are distinct, and $x_i + y_j \neq 0$ for all $1 \leq i, j \leq n$, it can be concluded that the determinant of matrix A is nonzero i.e. A is nonsingular. This is formalized in [GR13a] as follows.

Fact 2.8. *[GR13a] Consider distinct elements* $x_1, x_2, \ldots, x_n \in \mathbb{F}_{2^r}$ *and distinct elements* $y_1, y_2, \ldots, y_n \in \mathbb{F}_{2^r}$*, satisfying the condition that* $x_i + y_j \neq 0$ *for all* $1 \leq i, j \leq n$*. Then the Cauchy matrix* $A = (a_{i,j})$*, where* $1 \leq i, j \leq n$ *and* $a_{i,j} = \frac{1}{x_i + y_j}$*, is nonsingular.*

Fact 2.9. *[GR13a] Any square submatrix of a Cauchy matrix is again a Cauchy matrix.*

Definition 2.23. *(Vandermonde matrix) The matrix*

$$A = vand(x_1, x_2, x_3, \ldots, x_n) = \begin{bmatrix} 1 & 1 & 1 & \ldots & 1 \\ x_1 & x_2 & x_3 & \ldots & x_n \\ x_1^2 & x_2^2 & x_3^2 & \ldots & x_n^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_1^{n-1} & x_2^{n-1} & x_3^{n-1} & \ldots & x_n^{n-1} \end{bmatrix}$$

is called a Vandermonde matrix, where x_i's are elements of a finite or infinite field.

We sometimes use the notation $vand(\mathbf{x})$ to represent the Vandermonde matrix $vand(x_1, x_2, x_3, \ldots, x_n)$, where $\mathbf{x} = (x_1, x_2, x_3, \ldots, x_n)$.

It is known that

$$\det(vand(\mathbf{x})) = \prod_{1 \le i < j \le n} (x_j - x_i),$$

which is nonzero if and only if the x_i's are distinct.

Fact 2.10. *[MS77, Page 323] Any square submatrix of a Vandermonde matrix with real, positive entries is nonsingular, but this is not true over finite fields. For an example, consider*

$$vand(1, \alpha, \alpha^4, \alpha^5) = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & \alpha & \alpha^4 & \alpha^5 \\ 1 & \alpha^2 & \alpha^8 & \alpha^{10} \\ 1 & \alpha^3 & \alpha^{12} & \alpha^{15} \end{bmatrix}$$

where α is a primitive element of $\mathbb{F}_{2^4}$ constructed by the irreducible polynomial $x^4 + x + 1$. Consider the 2×2 submatrix

$$\begin{bmatrix} 1 & 1 \\ 1 & \alpha^{15} \end{bmatrix}$$

which is singular as $\alpha^{15} = 1$.

Consequently, these matrices themselves need not be MDS over a finite field. To address this, Lacan and Fimes [LF04a, LF04b] used two Vandermonde matrices to build an MDS matrix. We will provide a comprehensive discussion of Vandermonde based constructions in Chapter 3.

There are several generalizations of the Vandermonde matrices in the literature, as documented in [EM03, GKR78, KLK09, Pow67, Shp05, Van77] and the references therein. Our focus is on the variant presented in [KLK09], due to its applications in cryptography and error correcting codes. The definition of this variant is as follows.

Definition 2.24. *(Generalized Vandermonde matrix) Let* $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{F}_q^n$ *and* $T = \{t_1, t_2, \ldots, t_n\} \subset \mathbb{Z}$ *with* $0 \leq t_1 < t_2 < \ldots < t_n$*. Then the matrix*

$$V(\mathbf{x}, T) = \begin{bmatrix} x_1^{t_1} & x_2^{t_1} & \ldots & x_n^{t_1} \\ x_1^{t_2} & x_2^{t_2} & \ldots & x_n^{t_2} \\ \vdots & \vdots & \vdots & \\ x_1^{t_n} & x_2^{t_n} & \ldots & x_n^{t_n} \end{bmatrix}$$

is said to be a generalized Vandermonde matrix with respect to T*.*

Remark 2.11. *It can be observed that when* $T = \{0, 1, \ldots, n-1\}$*, the matrix* $V(\mathbf{x}, T)$ *corresponds to the Vandermonde matrix* $vand(\mathbf{x})$*.*

Let I denote the set of discontinuities in T, i.e., $I = \{0, 1, \ldots, t_n\} \setminus T$. Clearly, $0 \leq t_1 < t_2 < \ldots < t_n = n + |I| - 1$. Throughout the rest of the thesis, the notation $V_\perp(x; I)$ is used interchangeably with $V(\mathbf{x}; T)$.

Now we will see how the determinant of $V(\mathbf{x}; T)$ can be computed with the help of the determinant of a Vandermonde matrix when T has discontinuities. To do so, we require the following definition.

Definition 2.25. *The elementary symmetric polynomial of degree* d *can be expressed as:*

$$\sigma_d(x_1, x_2, \ldots, x_n) = \sum_{w(e)=d} x_1^{e_1} x_2^{e_2} \cdots x_n^{e_n},$$

where $e = (e_1, e_2, \ldots, e_n) \in \mathbb{F}_2^n$*.*

Theorem 2.8. *[KLK09, Theorem 1] If* $I = \{l_1, l_2, \ldots, l_s\}$*, we have*

$$\det(V_\perp(\mathbf{x}; I)) = \det(vand(\mathbf{x})) \det(S(\mathbf{x})),$$

where $S(\mathbf{x}) = (\sigma_{n-l_i+j-1}(\mathbf{x}))_{i,j=1}^s$*.*

Lemma 2.12. *[KLK09, Lemma 1] If* $I = \{l\}$*, we have*

$$\det(V_\perp(\mathbf{x}; I)) = \det(vand(\mathbf{x}))\sigma_{n-l}(\mathbf{x}).$$

By substituting $I = \{n-1\}$ and $I = \{1\}$ into Lemma 2.12, we can derive the Corollaries 2.10 and 2.11, respectively.

Corollary 2.10. *Let* $I = \{n-1\}$*, then* $\det(V_\perp(\mathbf{x}; I)) = \det(vand(\mathbf{x}))(\sum_{i=1}^n x_i)$*.*

Corollary 2.11. *Let* $I = \{1\}$ *and each* x_i *be a nonzero element of a field. Then we can express the determinant of* $V_\perp(\mathbf{x}; I)$ *as*

$$\det(V_\perp(\mathbf{x}; I)) = (\prod_{i=1}^{n} x_i) \det(vand(\mathbf{x}))(\sum_{i=1}^{n} x_i^{-1}).$$

Now, we will consider the case when T has more than one discontinuity, specifically, we will explore how to compute the determinant of $V_\perp(\mathbf{x}; I)$ when $I = \{1, n\}$.

Corollary 2.12. *Let $I = \{1, n\}$ and each x_i be a nonzero element of a field. Then we can express the determinant of $V_\perp(\mathbf{x}; I)$ as*

$$\det(V_\perp(\mathbf{x}; I)) = \det(vand(\mathbf{x})) \left(\prod_{i=1}^{n} x_i\right) \left[(\sum_{i=1}^{n} x_i)(\sum_{i=1}^{n} x_i^{-1}) - 1\right].$$

Proof. From Theorem 2.8, we know that

$$\det(V_\perp(\mathbf{x}; I)) = \det(vand(\mathbf{x})) \det(S(\mathbf{x})),$$

where $S(\mathbf{x}) = \begin{bmatrix} \sigma_{n-1}(\mathbf{x}) & \sigma_n(\mathbf{x}) \\ \sigma_0(\mathbf{x}) & \sigma_1(\mathbf{x}) \end{bmatrix}$. Thus, we have

$$\begin{aligned} \det(S(\mathbf{x})) &= \sigma_{n-1}(\mathbf{x})\sigma_1(\mathbf{x}) - \sigma_n(\mathbf{x})\sigma_0(\mathbf{x}) \\ &= \left[(\prod_{i=1}^{n} x_i \sum_{i=1}^{n} x_i^{-1})(\sum_{i=1}^{n} x_i)\right] - \prod_{i=1}^{n} x_i \\ &= \prod_{i=1}^{n} x_i \left[(\sum_{i=1}^{n} x_i)(\sum_{i=1}^{n} x_i^{-1}) - 1\right]. \end{aligned}$$

Therefore, $\det(V_\perp(\mathbf{x}; I)) = \det(vand(\mathbf{x})) \, (\prod_{i=1}^{n} x_i) \left[(\sum_{i=1}^{n} x_i)(\sum_{i=1}^{n} x_i^{-1}) - 1\right]$. □

Cauchy matrices are always MDS, meaning that it is not possible to obtain NMDS matrices directly from them. Furthermore, there is currently no known construction method for NMDS matrices using Vandermonde matrices. However, NMDS matrices can be constructed using generalized Vandermonde matrices. Other matrix structures, such as *Hadamard, circulant, left-circulant, Toeplitz*, and *Hankel* matrices, can also be utilized to construct both nonrecursive MDS and NMDS matrices. The benefit of using circulant, left-circulant, and Hadamard matrices is that they contain at most n distinct elements for an $n \times n$ matrix, which reduces the search space when employing these matrices in the search methods for finding MDS or NMDS matrices.

Definition 2.26. *(Circulant matrix) An $n \times n$ matrix A is said to be a right circulant (or circulant) matrix if its elements are determined by the elements of its first row*

$x_1, x_2, \ldots, x_n$ *as*

$$A = Circ(x_1, x_2, \ldots, x_n) = \begin{bmatrix} x_1 & x_2 & \ldots & x_n \\ x_n & x_1 & \ldots & x_{n-1} \\ \vdots & \vdots & \vdots & \vdots \\ x_2 & x_3 & \ldots & x_1 \end{bmatrix}.$$

In AES [DR02], the circulant MDS matrix used is $Circ(\alpha, 1+\alpha, 1, 1)$, where α is the root of $x^8 + x^4 + x^3 + x + 1$. The cipher WIDEA [JM09] employs an 8×8 circulant MDS matrix, $Circ(1, 1, \alpha^2, 1, \alpha^3, 1+\alpha^2, \alpha, 1+\alpha^3)$, where α is the root of $x^{16}+x^5+x^3+x^2+1$. On the other hand, the lightweight block ciphers Midori [BBI+15] and MANTIS [BJK+16] use a circulant NMDS matrix of order 4, $Circ(0, 1, 1, 1)$.

Such matrices find their applications in cryptography mainly due to the repetition of entries. For example, the AES diffusion matrix has two 1's in its first row, and since 1 is the multiplicative identity, it has implementation advantages as multiplication by 1 requires no implementation cost. Also, circulant matrices offer the advantage of being adaptable for implementation in both round-based and serialized implementations [DR02].

However, to the best of our knowledge, till date there is no known method to provide circulant matrix of arbitrary order which is MDS or NMDS by construction itself. Circulant MDS (or NMDS) matrices are obtained by search technique. Though search methods provide efficient MDS (or NMDS) matrices of moderate order over moderate size search space, it fails for higher order and large search space.

It is worth noting that the circulant MDS matrix $Circ(\alpha, 1+\alpha, 1, 1)$ used in the AES MixColumn operation has elements with low Hamming weights. However, this matrix contains a total number of 8 ones. In the paper [JV05b] by Junod et al., it was demonstrated that the maximum number of ones in a 4×4 MDS matrix is 9. To achieve this maximum, they introduced a new class of efficient MDS matrices, where the submatrices are circulant matrices. In another study by Gupta et al. [GR15], these newly discovered matrices were formally defined as *Type-I circulant-like* matrices. They conducted an extensive study of such matrices in the context of constructing efficient and perfect diffusion layers. Here, we present the definition of *Type-I circulant-like* matrices as provided in the works of [GR15, JV05b].

Definition 2.27. *(*Type-I circulant-like matrix*)[GR15, JV05b] The $n \times n$ matrix*

given by

$$\begin{bmatrix} a & \mathbf{1} \\ \mathbf{1}^T & A \end{bmatrix}$$

is referred to as a Type-I circulant-like matrix, where $A = Circ(1, x_2, \ldots, x_{n-1})$, $\mathbf{1} = \underbrace{(1, \ldots, 1)}_{n\text{-}1 \text{ times}}$, *1 is the unit element and* x_i*'s and a are any nonzero elements of the underlying field other than* 1. *This matrix is denoted as* $TypeI(a, Circ(1, x_2, \ldots, x_{n-1}))$.

The inverse of an MDS matrix is of particular importance in the context of SPN networks. In [GR15], it was noted that the inverses of matrices belonging to the class of Type-I circulant-like matrices exhibit a similar structure. With this observation in mind, the following definition is introduced.

Definition 2.28. *(*AlmostType-I circulant-like *matrix)[GR15] The* $n \times n$ *matrix given by*

$$\begin{bmatrix} a & \mathbf{b} \\ \mathbf{b}^T & A \end{bmatrix}$$

is referred to as an AlmostType-I circulant-like matrix, where $A = Circ(x_1, x_2, \ldots, x_{n-1})$, $\mathbf{b} = \underbrace{(b, \ldots, b)}_{n\text{-}1 \text{ times}}$ *and a, b and* x_i*'s can be any elements from the underlying field. This matrix is denoted as* $AlmostTypeI(a, b, Circ(x_1, \ldots, x_{n-1}))$.

Example 2.6. *Consider the* Type-*I* circulant-like *matrix* $A = TypeI(\alpha, Circ(1, 1 + \alpha + \alpha^{-1}, \alpha))$ *over* $\mathbb{F}_{2^8}$ *whose constructing polynomial is* $x^8+x^7+x^6+x^5+x^4+x^3+1$ *and* α *is a root of that polynomial. Then the inverse of A is a* AlmostType-I circulant-like *matrix, where* $A^{-1} = AlmostTypeI(\alpha^7 + \alpha^6 + \alpha^5 + \alpha^4 + \alpha^3 + \alpha^2 + 1, \alpha, Circ(1, \alpha^7 + \alpha^6 + \alpha^4, \alpha^7 + \alpha^6 + \alpha^4 + \alpha^2 + 1))$.

It is worth noting that in the context of SPN networks, involutory or orthogonal MDS matrices are highly desirable. It will be shown in Lemma 3.17 that circulant matrices cannot be both involutory and MDS. In Lemma 3.14 it will also be shown that $2^n \times 2^n$ circulant matrix cannot be both MDS and orthogonal. Again Lemma 3.18 to Lemma 3.21 shows that *Type-I circulant-like* matrices are neither involutory nor orthogonal. In order to address this, a novel class of involutory circulant-like MDS matrices was introduced in [GR15]. This construction was derived from a scheme proposed in a previous work [YMT97], where the authors focused on constructing involutory MDS matrices of size $2n \times 2n$ based on an initial MDS submatrix of size $n \times n$. In the study [GR15], the MDS submatrices were chosen to be circulant matrices,

resulting in the establishment of a new type of circulant-like matrices. The following definition captures this construction.

Definition 2.29. *(Type-II circulant-like matrix)[GR15] The $2n \times 2n$ matrix given by*

$$\begin{bmatrix} A & A^{-1} \\ A^3 + A & A \end{bmatrix}$$

is referred to as Type-II circulant-like matrix, where $A = Circ(x_1, \ldots, x_n)$. This matrix is denoted as $TypeII(Circ(x_1, \ldots, x_n))$.

In [LS16], the authors suggest a new category of matrices known as *left-circulant matrices*. These matrices retain the advantages of circulant matrices.

Definition 2.30. *(Left-circulant matrix) An $n \times n$ matrix A is said to be a left-circulant matrix if each successive row is obtained by a left shift of the previous row i.e.*

$$A = l\text{-}Circ(x_1, x_2, \ldots, x_n) = \begin{bmatrix} x_1 & x_2 & \ldots & x_n \\ x_2 & x_3 & \ldots & x_1 \\ \vdots & \vdots & \vdots & \vdots \\ x_n & x_1 & \ldots & x_{n-1} \end{bmatrix}.$$

It is important to note that a left-circulant matrix is symmetric. Therefore, if the matrix is orthogonal, it is also involutory, and vice versa.

Remark 2.12. *It is worth noting that a left-circulant matrix is a row-permuted circulant matrix. More specifically, for $A = Circ(x_1, x_2, \ldots, x_n)$, we have $PA = l\text{-}Circ(x_1, x_2, \ldots, x_n)$, where P is the permutation matrix given by*

$$P = \begin{bmatrix} 1 & 0 & 0 & \ldots & 0 & 0 & 0 \\ 0 & 0 & 0 & \ldots & 0 & 0 & 1 \\ 0 & 0 & 0 & \ldots & 0 & 1 & 0 \\ \vdots & \vdots & \vdots & \ldots & \vdots & \vdots & \vdots \\ 0 & 1 & 0 & \ldots & 0 & 0 & 0 \end{bmatrix}. \tag{2.1}$$

Further discussion of constructing MDS and NMDS matrices from circulant and left-circulant matrices will be presented later.
The authors in [SDMO12] introduced a particular type of matrices over $\mathbb{F}_{2^r}$ referred to as Finite Field Hadamard (FFHadamard) matrices. In [BR00a, PSA+18, SKOP15], authors call the FFHadamard matrix as the *Hadamard matrix*. In this thesis, we also call it Hadamard.

Definition 2.31. *(Hadamard matrix) A $2^n \times 2^n$ matrix H is Hadamard matrix in $\mathbb{F}_{2^r}$ if it can be expressed in the form:*

$$\begin{bmatrix} U & V \\ V & U \end{bmatrix}$$

where the submatrices U and V are also Hadamard matrices.

For example a $2^2 \times 2^2$ Hadamard matrix is:

$$H = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 \\ x_2 & x_1 & x_4 & x_3 \\ x_3 & x_4 & x_1 & x_2 \\ x_4 & x_3 & x_2 & x_1 \end{bmatrix}.$$

Remark 2.13. *Noting that Hadamard matrices commute and since we are working in a field of characteristic 2, it is easy to check by induction that $H^2 = c^2 I$, where c is the sum of the elements of the first row. Therefore, if the sum of the elements of the first row is equal to 1, then it will be an involutory matrix.*

It is worth noting that the Anubis block cipher [BR00a] uses a Hadamard involutory MDS matrix, whose first row is $(1, \alpha, \alpha^2, \alpha + \alpha^2)$, where α is the root of the primitive polynomial $x^8 + x^4 + x^3 + x^2 + 1$. Also, the block ciphers Khazad [BR00b] and CLEFIA [SSA+07] also use Hadamard involutory MDS matrices in their diffusion layers.

Toeplitz matrices have a deep interconnection with circulant matrices, and the interested reader may consult [Gra06, GKPS04] for more information about the connection.

Definition 2.32. *(Toeplitz matrix) The $n \times n$ matrix*

$$A = \begin{bmatrix} x_1 & x_2 & x_3 & \dots & x_{n-1} & x_n \\ y_1 & x_1 & x_2 & \dots & x_{n-2} & x_{n-1} \\ y_2 & y_1 & x_1 & \dots & x_{n-3} & x_{n-2} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ y_{n-1} & y_{n-2} & y_{n-3} & \dots & y_1 & x_1 \end{bmatrix}$$

is called a Toeplitz matrix of order n.

A Toeplitz matrix is a special kind of matrix in which every descending diagonal from left to right is constant. Also, it is easy to check that circulant matrices are a special type of Toeplitz matrices.

A Toeplitz matrix can be defined based on the elements in its first row and first column. For instance $\{x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_{n-1}\}$ defines the Toeplitz matrix A of Definition 2.32 and in this thesis we will use the notation $Toep(x_1, x_2, \ldots, x_n;\ y_1, y_2, \ldots, y_{n-1})$ for describing A.

Hankel matrices, introduced in [GPRS19] for MDS matrix construction, are similar to Toeplitz matrices in that each ascending skew diagonal from left to right is constant.

Definition 2.33. *(Hankel matrix) The $n \times n$ matrix*

$$A = \begin{bmatrix} x_1 & x_2 & x_3 & \ldots & x_{n-1} & x_n \\ x_2 & x_3 & x_4 & \ldots & x_n & y_1 \\ x_3 & x_4 & x_5 & \ldots & y_1 & y_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n & y_1 & y_2 & \ldots & y_{n-2} & y_{n-1} \end{bmatrix}$$

is called a Hankel matrix.

Note that a left-circulant matrix is a special case of a Hankel matrix. A Hankel matrix is symmetric and is defined by its first row and last column. For instance $\{x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_{n-1}\}$ defines the Hankel matrix A of Definition 2.33 and in this thesis we will use the notation $Hank(x_1, x_2, \ldots, x_n;\ y_1, y_2, \ldots, y_{n-1})$ for describing A. Also note that since the Hankel matrix is symmetric, an involutory (orthogonal) Hankel matrix is orthogonal (involutory).
Till now, we have presented matrix structures that are used to construct nonrecursive MDS or NMDS matrices. Now we will mention some sparse matrix structures that are used in the recursive approach.

2.5.2 Matrix structures for recursive approaches

The advantage of using *recursive MDS* or *NMDS* matrices is their suitability for lightweight implementations. The implementation of the diffusion layer can be achieved through the recursive execution of sparse matrices, which typically requires a small number of clock cycles. This approach, based on *companion matrices*, has

been utilized in the PHOTON family of hash functions [GPP11] and the LED block cipher [GPPR11] due to their simplicity in implementation using a simple LFSR.

Definition 2.34. *(Companion matrix) Consider a monic polynomial* $g(x) = a_1 + a_2x + \ldots + a_nx^{n-1} + x^n \in \mathbb{F}_q[x]$ *of degree* n*. Then, the companion matrix* $C_g \in M_n(\mathbb{F}_q)$ *corresponding to the polynomial* g *is defined as follows:*

$$C_g = \begin{bmatrix} 0 & 1 & 0 & \ldots & 0 \\ \vdots & & \ddots & & \vdots \\ 0 & 0 & \ldots & \ldots & 1 \\ -a_1 & -a_2 & \ldots & \ldots & -a_n \end{bmatrix}.$$

We sometimes use the notation $\mathsf{Companion}(-a_1, -a_2, \ldots, -a_n)$ to represent the companion matrix C_g. Observe that if $a_1 \neq 0$ then the matrix C_g is nonsingular and its inverse is given by

$$C_g^{-1} = \begin{bmatrix} \frac{-a_2}{a_1} & \frac{-a_3}{a_1} & \ldots & \frac{-a_n}{a_1} & \frac{-1}{a_1} \\ 1 & 0 & \ldots & 0 & 0 \\ \vdots & \ddots & & \vdots & \vdots \\ 0 & 0 & \ldots & 1 & 0 \end{bmatrix}.$$

It should be noted that when $a_1 = 1$, the elements in the first row of C_g^{-1} are equal to the elements in the last row of C_g. In fact, in this case, we have

$$C_g^{-1} = PC_gP, \tag{2.2}$$

where

$$P = \begin{bmatrix} 0 & 0 & \ldots & 0 & 1 \\ 1 & 0 & \ldots & 0 & 0 \\ \vdots & \vdots & \ddots & & \vdots \\ 0 & 0 & \ldots & 1 & 0 \end{bmatrix}$$

is a permutation matrix.

Definition 2.35. *A square matrix* $M \in M_n(\mathbb{F}_q)$ *is said to be diagonalizable if* M *is similar to a diagonal matrix. This means* $M = PDP^{-1}$ *for some diagonal matrix* D *and a nonsingular matrix* P*.*

Now, we will consider some results related to diagonalizable companion matrices.

Lemma 2.13. *[GPV15] Let* $C_g \in M_n(\mathbb{F}_q)$ *be a nonsingular companion matrix which is diagonalizable, say* $C_g = PDP^{-1}$ *where* P *is a nonsingular matrix of order* n *and*

$D = diag(\lambda_1, \lambda_2, \ldots, \lambda_n)$. Then all entries of P will be nonzero. Moreover, C_g can be expressed as $C_g = VDV^{-1}$, where $V = vand(\lambda_1, \lambda_2, \ldots, \lambda_n)$.

Corollary 2.13. *[GPV15] A companion matrix C_g is nonsingular and diagonalizable if and only if all eigenvalues of C_g are distinct and nonzero.*

Lemma 2.14. *[RB00] If M is an $n \times n$ matrix with n distinct eigenvalues, then M is diagonalizable.*

Theorem 2.9. *[RB00] The characteristic polynomial of C_g, as defined in Definition 2.34, is the polynomial $g(x) = a_1 + a_2x + \ldots + a_nx^{n-1} + x^n$.*

Since the roots of a characteristic polynomial are the eigenvalues, based on Lemma 2.13, Lemma 2.14 and Theorem 2.9, we can conclude the following result for a companion matrix.

Theorem 2.10. *If the monic polynomial $g(x) = a_1 + a_2x + \ldots + a_nx^{n-1} + x^n$ has n distinct roots $\lambda_1, \lambda_2, \ldots, \lambda_n$, then C_g can be expressed as $C_g = VDV^{-1}$, where $V = vand(\lambda_1, \lambda_2, \ldots, \lambda_n)$ and $D = diag(\lambda_1, \lambda_2, \ldots, \lambda_n)$.*

In [TTKS18], authors introduce a new kind of sparse matrices known as *DSI matrices* for the construction of recursive MDS matrices.

Definition 2.36. *(DSI matrix)[TTKS18] Let $a = (a_1, a_2, \ldots, a_n)$ and $b = (b_1, b_2, \ldots, b_{n-1})$ where $a_i, b_j \in \mathbb{F}_{2^r}$ for $1 \leq i \leq n$ and $1 \leq j \leq n-1$. A Diagonal-Serial-Invertible (DSI) matrix M is determined by two vectors a and b defined as follows:*

$$(M)_{i,j} = \begin{cases} a_1, & i = 1, j = n \\ a_i, & i = j + 1 \\ b_i, & i = j \leq n - 1 \\ 0 & otherwise. \end{cases}$$

Definition 2.37. *(Sparse DSI matrix)[TTKS18] A DSI matrix $M = DSI(a_1, a_2, \ldots, a_n; b_1, b_2, \ldots, b_{n-1})$ of order n is sparse if it satisfies:*

$$\begin{cases} b_2 = b_4 = \ldots = b_{n-2} = 0, & \text{if } n \text{ is even} \\ b_2 = b_4 = \ldots = b_{n-3} = 0, & \text{if } n \text{ is odd.} \end{cases}$$

Example 2.7. *An example of a sparse DSI matrix of order 4 and 5 are given below:*

$$\begin{bmatrix} b_1 & 0 & 0 & a_1 \\ a_2 & 0 & 0 & 0 \\ 0 & a_3 & b_3 & 0 \\ 0 & 0 & a_4 & 0 \end{bmatrix}, \qquad \begin{bmatrix} b_1 & 0 & 0 & 0 & a_1 \\ a_2 & 0 & 0 & 0 & 0 \\ 0 & a_3 & b_3 & 0 & 0 \\ 0 & 0 & a_4 & b_4 & 0 \\ 0 & 0 & 0 & a_5 & 0 \end{bmatrix}.$$

2.6 Hardware Cost of a Diffusion matrix

The efficiency of hardware implementation in a given operation can be measured in terms of the amount of area required. The area is a critical factor in hardware design, and optimizing it can lead to more compact and cost-effective implementations. This metric is particularly important in fields such as integrated circuit design, where minimizing the physical footprint of the hardware can have significant implications for performance and cost.

In the past, it was widely believed that implementing the multiplication of finite field elements with low Hamming weights incurred lower hardware costs. As of 2014, the authors of [KPPY14] proposed an approach to estimate implementation costs by counting the number of XOR gates (*d-XOR gates*) required to implement the field element based on the multiplicative matrix of the element. According to them, higher Hamming weight elements can be implemented at a lower cost than previously thought. To better estimate the cost of hardware implementation, the authors of [JPST17] suggested a metric called *s-XOR*.

2.6.1 XOR count

It is important to emphasize that the diffusion matrix can solely be implemented utilizing XOR gates, which leads to the subsequent definition.

Definition 2.38. *[Köl19] The d-XOR count (direct XOR count) of a matrix $A \in GL(r, \mathbb{F}_2)$, represented as d-XOR(A), is determined by*

$$d\text{-}XOR(A) = wt(A) - r,$$

where $wt(A)$ represents the number of ones in the matrix A.

Remark 2.14. *It is important to emphasize that the d-XOR count of an invertible matrix is always non-negative, as every row of an invertible matrix must contain at*

least one nonzero element. Also, d-XOR(A) is zero if and only if A is a permutation matrix.

Example 2.8. *Consider the matrix*

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} \in GL(4, \mathbb{F}_2).$$

Then we have

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix} = \begin{bmatrix} v_1 \\ v_1 + v_2 \\ (v_1 + v_2) + v_3 \\ ((v_1 + v_2) + v_3) + v_4 \end{bmatrix}.$$

Thus, d-XOR(A) $= 10 - 4 = 6$.

It can be easily verified that the multiplication with A can be performed using only 3 XOR operations, as the outputs from the previous steps can be reused. This is facilitated by a metric known as s-XOR, which is introduced in [JPST17].

Definition 2.39. *[Köl19] The s-XOR count (sequential XOR count) of a matrix* $A \in GL(r, \mathbb{F}_2)$*, denoted by s-XOR(A), refers to the minimum non-negative integer* t *for which* A *can be expressed as*

$$A = P \prod_{k=1}^{t} (I_r + E_{i_k, j_k}),$$

where P *is a permutation matrix and* E_{i_k, j_k}*, with* $i_k \neq j_k$ *for all* k*, is a binary matrix with 1 as* (i_k, j_k)*-th entry and 0 elsewhere.*

Example 2.9. *In Example 2.8, it is straightforward to identify an s-XOR representation that requires only 3 XOR operations.*

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} = I_4(I_4 + E_{4,3})(I_4 + E_{3,2})(I_4 + E_{2,1}).$$

We can observe that we require a minimum of three addition matrices (i.e. $I_r + E_{i_k,j_k}$) because all rows except the first one require at least one update. Thus, the s-XOR representation presented earlier is optimal, and it has s-XOR$(A) = 3$.

When a basis of $\mathbb{F}_{2^r}$ is given, the multiplication of $\alpha \in \mathbb{F}_{2^r}$ given by $x \to \alpha x$ can be expressed as the multiplication of a matrix in $GL(r, \mathbb{F}_2)$. The matrix depends not only on α, but also on the choice of basis of $\mathbb{F}_{2^r}$ over $\mathbb{F}_2$. Let $M_{\alpha,B}$ be the matrix representation of the mapping $x \to \alpha x$ with respect to the basis B.

Definition 2.40. *[Köl19] Let $\alpha \in \mathbb{F}_{2^r}$. Then the d-XOR count and s-XOR count of α, denoted by d-XOR(α) and s-XOR(α) respectively, are given by*

$$\text{d-XOR}(\alpha) = \min_B \text{d-XOR}(M_{\alpha,B}) \quad \text{and} \quad \text{s-XOR}(\alpha) = \min_B \text{s-XOR}(M_{\alpha,B}),$$

where the minimum is taken over all bases of $\mathbb{F}_{2^r}$ over $\mathbb{F}_2$.

The d-XOR count (s-XOR count) of $M_{\alpha,B}$ generally differs from the d-XOR count (s-XOR count) of $M_{\alpha,B'}$ for different bases B and B'. Generally, it is not easy to determine the s-XOR count of a given field element. An exhaustive search is conducted in [BKL16] to find matrices that optimize the s-XOR count metric. As a result, the s-XOR count and an optimal matrix representation for every element $\alpha \in \mathbb{F}_{2^r}$ for $r \leq 8$ are found. For more details about the two XOR count metrics, see [Köl19] and the related references therein.

Throughout the thesis, we denote the XOR count of $\alpha \in \mathbb{F}_{2^r}$ as XOR(α), which can either be the d-XOR count or the s-XOR count, unless specified otherwise.

Fixed XOR of a matrix: The cost of implementing a diffusion matrix can be determined by adding up the XOR counts of each entry in the matrix. If a row has k_i nonzero elements from the field $\mathbb{F}_{2^r}$, these k_i elements must be combined, which incurs a fixed XOR cost of $(k_i - 1)r$. Therefore, if an n order matrix has $k_1, k_2, \ldots, k_n$ nonzero elements in its n rows, the matrix incurs a fixed XOR cost of $\sum_{i=1}^{n}(k_i - 1)r$ in $\mathbb{F}_{2^r}$.

The sum, $\mathcal{K} = \sum_{i=1}^{n}(k_i - 1)$, is referred to as the fixed XOR of the matrix. For example, an MDS matrix of order n has a fixed XOR of $\mathcal{K} = n(n-1)$, while for a companion and DSI matrix of order n, it is $n - 1$, and for a sparse DSI matrix of order n, it is $\lceil n/2 \rceil$.

XOR count of a matrix: The XOR count of an n order matrix M, denoted by $XOR(M)$, over the field $\mathbb{F}_{2^r}$ is calculated as $\sum_{i,j=1}^{n} XOR((M)_{i,j}) + \mathcal{K} \cdot r$, where

$XOR((M)_{i,j})$ is the XOR count of the entry $(M)_{i,j}$ in M.

Global optimization: It is worth mentioning that recently a lot of attention has been paid to the search for efficiently implementable MDS matrices by global optimization techniques. For instance see [BFI19, DL18, KLSW17, LSL+19, YZW21]. So we can categorize the search methods into two categories: local optimization and global optimization.

In local optimizations, designers mainly focus on the selection of matrix entries with low d-XOR or s-XOR counts. Whereas in global optimizations, given a matrix A, we can obtain an estimation of its hardware cost by finding a good linear straight-line program corresponding to A with state-of-the-art automatic tools based on a certain SLP heuristic [BMP12]. A globally optimized implementation can be significantly cheaper than the local optimization because common intermediate values can be computed once and reused. However, this thesis focuses on the local optimization technique to calculate the implementation cost of the matrices.

So far, we have discussed various matrix structures used as diffusion matrices. Next, we will explore nonlinear functions, such as Sboxes and Boolean functions, as they play a vital role in achieving the confusion property.

2.7 Boolean Functions and Sboxes

A *Boolean function* g with n variables is a mapping from $\mathbb{F}_2^n$ to $\mathbb{F}_2$. The support of a Boolean function g, denoted by $Sup(g)$, is defined as the set of inputs x for which $g(x) = 1$ i.e., $Sup(g) = \{x : g(x) = 1\}$.

The *weight of a Boolean function* g, denoted by $w(g)$, is the cardinality of its support, i.e., the number of inputs x such that $g(x) = 1$. A function g is considered balanced if its weight is equal to 2^{n-1}.

A Boolean function can be represented by its binary output vector containing 2^n elements, referred to as the truth table. Another way of representing g is by its algebraic normal form:

$$g(x) = \bigoplus_{(\alpha_n,\alpha_{n-1},\ldots,\alpha_2,\alpha_1)\in\mathbb{F}_2^n} A_g(\alpha_n,\alpha_{n-1},\ldots,\alpha_2,\alpha_1)x_n^{\alpha_n}x_{n-1}^{\alpha_{n-1}}\ldots x_2^{\alpha_2}x_1^{\alpha_1}$$

where $x = (x_n, x_{n-1}, \ldots, x_2, x_1) \in \mathbb{F}_2^n$ and A_g is a Boolean function on $\mathbb{F}_2^n$.

The nonlinearity of a Boolean function is a key parameter in cryptography. It measures the Hamming distance[1] of an n-variable Boolean function from the set of all affine functions. Let A_n be the set of all n-variable affine functions. Then, the *nonlinearity of the Boolean function* g is defined as

$$nl(g) = \min_{l \in A_n} d(g, l).$$

The maximum achievable nonlinearity for an n-variable Boolean function is given by $2^{n-1} - 2^{(n-2)/2}$. Boolean functions that attain this maximum nonlinearity are referred to as bent functions. It is important to note that bent functions can only exist when n is an even number [Rot76].

Definition 2.41. *An $n \times m$ Sbox is a mapping $S : \mathbb{F}_2^n \to \mathbb{F}_2^m$.*

Then, to each $x = (x_n, x_{n-1}, \ldots, x_2, x_1) \in \mathbb{F}_2^n$ some $y = (y_m, y_{m-1}, \ldots, y_2, y_1) \in \mathbb{F}_2^m$ is assigned by $S(x) = y$. The $n \times m$ Sbox S can be considered as a vectorial Boolean function comprising m individual Boolean functions $f_m, f_{m-1}, \ldots, f_2, f_1 : \mathbb{F}_2^n \to \mathbb{F}_2$, where $f_i(x) = y_i$ for $i = 1, 2, \ldots, m$. These functions are referred to as the coordinate Boolean functions of the Sbox. Thus, we can write

$$S(x) = (f_m(x), f_{m-1}(x), \ldots, f_2(x), f_1(x)).$$

It is well-known that most of the desirable cryptographic properties of the Sbox can also be defined in terms of all nontrivial linear combinations of the coordinate functions, referred to as the Sbox component Boolean functions $g_c : \mathbb{F}_2^n \to \mathbb{F}_2$, where

$$g_c = c_m f_m \oplus \ldots \oplus c_2 f_2 \oplus c_1 f_1 \text{ and } c = (c_m, \ldots, c_2, c_1) \in \mathbb{F}_2^m \setminus \{0\}.$$

To avoid trivial statistical attacks, an Sbox should be regular (balanced). An $n \times m$ Sbox S with $n \geq m$ is said to be regular if, for each its output $y \in \mathbb{F}_2^m$, there are exactly 2^{n-m} inputs that are mapped to y. Clearly, each bijective $n \times n$ Sbox S is always regular since it represents a permutation. It is well-known that an $n \times m$ Sbox with $n \geq m$ is regular if and only if all its component Boolean functions are balanced [SZZ94].

The *nonlinearity of an Sbox* is a fundamental parameter in cryptography. It is determined by considering the minimum nonlinearity among the nonlinearities of its

[1] The Hamming distance $d(f, g)$ between two functions f and g, defined on a same set A, is defined to be the size of $\{x \in A : f(x) \neq g(x)\}$.

component Boolean functions. The nonlinearity of the Sbox S is expressed as:

$$nl(S) = \min_{c \in \mathbb{F}_2^m \setminus \{0\}} nl(g_c).$$

The best known nonlinearity of a 4-variable balanced Boolean function is 4 [Car21, Table 3.2]. Thus, the maximum nonlinearity of an 4×4 bijective Sbox is 4.
In this thesis, we are discussing about $n \times n$ bijective Sboxes, and we will call these as n-bit Sboxes.
Differential cryptanalysis [BS91a] is a method that analyzes the impact of specific differences in plaintext pairs on the resulting differences in ciphertexts. Suppose that two values x and x', with a difference Δx (i.e. $\Delta x = x \oplus x'$) are processed by a function F. Let Δy be the output difference i.e. $\Delta y = F(x) \oplus F(x')$. In this context, we say that Δx *propagates* to Δy through the function F, denoted by $\Delta x \xrightarrow{F} \Delta y$. The pair $(\Delta x, \Delta y)$ is then referred to as a *differential* over F.

Definition 2.42. *For a vectorial function $F\colon \mathbb{F}_2^n \mapsto \mathbb{F}_2^n$ and any two vectors $a, b \in \mathbb{F}_2^n$, let us define*

$$\delta_F(a, b) = \#\{x \in \mathbb{F}_2^n : F(x \oplus a) \oplus F(x) = b\}.$$

Then, the differential probability of the differential (a, b) over F is defined by

$$\Pr[a \xrightarrow{F} b] = \frac{\delta_F(a, b)}{2^n}.$$

Definition 2.43. *For an n-bit Sbox S, the difference distribution table (DDT) of S is the table of size $2^n \times 2^n$, with rows and columns indexed by input and output differences respectively. The corresponding entries are equal to the integers $\delta_S(a, b)$ for the particular differential (a, b).*

The *differential uniformity* of S, denoted by δ_S, is the highest value in the DDT, i.e. $\delta_S = \max_{a,b \in \mathbb{F}_2^n, a \neq 0} \delta_S(a, b)$ and $\frac{\delta_S}{2^n}$ is called the *maximal differential probability* of the Sbox S. In Section 2.8.4, we will see that an Sbox should have low differential uniformity to increase block cipher immunity to differential cryptanalysis

In the realm of *linear cryptanalysis* [Mat94], a linear approximation is commonly expressed in terms of vectors (also known as masks) a, b.

Definition 2.44. *For a vectorial function $F\colon \mathbb{F}_2^n \mapsto \mathbb{F}_2^n$, a linear approximation is defined as a tuple (a, b) with $a, b \in \mathbb{F}_2^n$. Let us define*

$$L_F(a,b) = \#\{x \in \mathbb{F}_2^n : x \cdot a \oplus F(x) \cdot b = 0\} - 2^{n-1},$$

where '$\cdot$' denotes the bitwise logical AND. Then the bias of the linear approximation (a,b) is defined as

$$\epsilon_F(a,b) = \frac{L_F(a,b)}{2^n}$$

and its correlation is defined as

$$Cor(a \xrightarrow{F} b) = 2 \cdot \epsilon_F(a,b).$$

Definition 2.45. *For an n-bit Sbox S, the linear approximation table (LAT) of S is the table of size $2^n \times 2^n$, with rows and columns indexed by input and output masks respectively. The corresponding entries are equal to the integers $L_S(a,b)$ for the particular linear approximation (a,b).*

The *maximal absolute bias* of a linear approximation of an Sbox is given by $\frac{L_S}{2^n}$, where $L_S = \max_{a,b \in \mathbb{F}_2^n, a \neq 0} |L_S(a,b)|$. In Section 2.8.4, we will see that, like the differential uniformity, a lower value of L_S is needed to enhance the block cipher's resistance to linear cryptanalysis [Mat94].

2.8 Block cipher

Since modern ciphers are designed to operate on computers, it is generally assumed that messages and keys are encoded as binary vectors. A *block cipher* converts plaintext blocks with a fixed length of n and transforms them into ciphertext blocks with the same length, using a secret key K.

Definition 2.46. *A block cipher $E(P,K)$ is a function from $\mathbb{F}_2^n \times \mathbb{F}_2^k$ to $\mathbb{F}_2^n$ with the property that, for each key $K \in \mathbb{F}_2^k$, $E(P,K)$ is a permutation on $\mathbb{F}_2^n$.*

We refer to the parameter n as its block length and to k as its key length. For $P \in \mathbb{F}_2^n$ and $C = E(P,K)$, we call P the *message or plaintext*, and C the *ciphertext* corresponding to encryption under the key K. Decrypting a ciphertext $C \in \mathbb{F}_2^n$ back to the plaintext P is done through $P = E^{-1}(C,K)$. Modern ciphers commonly employ a block length of $n = 64, 128$, or 256 bits. In order to encrypt messages of arbitrary lengths, block ciphers are combined with a mode of operation such as CBC, CFB, OFB [Dwo01], etc.

A block cipher with a key size of k has 2^k possible keys, and each key defines a permutation of 2^n inputs. There are $(2^n)!$ different permutations on n-bit input

blocks which, by using Stirling's approximation, is approximately $2^{(n-1)2^n}$. Although for typical values of n and k a block cipher covers only a small portion of all possible permutations, a secure block cipher is expected to hide this fact. In simpler terms, when we randomly choose a key, it should appear as if the permutation is selected randomly from the vast number of possibilities.

In the design of block ciphers, a crucial aspect is the design of a round function. Broadly, there are two main frameworks for this purpose: the *substitution-permutation network* and the *feistel network*.

2.8.1 Substitution-Permutation Network (SPN)

A substitution-permutation network (SPN) defines a unique structure for the round functions in an iterative cipher. In this configuration, each round involves two main operations: a nonlinear operation (substitution layer), represented by a parallel application of smaller nonlinear functions, and a linear transformation (permutation layer or linear layer).

The SPN structure was introduced in [FNS75], where the linear layer, defined as a bit permutation- a matrix associated with L is a permutation matrix over $\mathbb{F}_2$. Nowadays, most SPN ciphers follow the concept of a key-alternating cipher, where the unkeyed round functions can be broken down into an invertible nonlinear layer and an invertible linear layer. It is crucial to note that both the linear and nonlinear layer must be invertible because the decryption process in SPN cipher involves reversing both the substitution and permutation layer. The structure is depicted in Figure 2-1.

In a more formal way, the nonlinear substitution layer $\mathcal{N}\colon \mathbb{F}_2^n \mapsto \mathbb{F}_2^n$ is defined by applying a b-bit Sbox S in parallel n_b times, where $n = n_b \cdot b$. It is important to note that since $n = n_b \cdot b$, $\mathbb{F}_2^n$ is isomorphic to $\underbrace{\mathbb{F}_2^b \times \mathbb{F}_2^b \times \cdots \times \mathbb{F}_2^b}_{n_b \text{ times}}$. Therefore, $\mathcal{N}\colon \mathbb{F}_2^b \times \mathbb{F}_2^b \times \cdots \times \mathbb{F}_2^b \to \mathbb{F}_2^b \times \mathbb{F}_2^b \times \cdots \times \mathbb{F}_2^b$ given by

$$\mathcal{N}(x_1, x_2, \ldots, x_{n_b}) = (S(x_1), S(x_2), \cdots, S(x_{n_b})).$$

Generally, a small value is chosen for b. Common choices include $b = 4$ or $b = 8$. It is important to note that, in the substitution layer, one can use a different Sbox for each position. However, for simplicity in implementation, it is common to use the same Sbox in parallel.

The linear layer (L), can be characterized by a matrix $M \in GL(n, \mathbb{F}_2)$. Nowadays,

in an SPN cipher, L is permitted to be any invertible linear transformation, not necessarily a bit permutation. The adoption of a general linear layer over a bit permutation was largely influenced by the *wide-trail strategy* [Dae95]. While a bit permutation might offer a simpler implementation, the wide-trail strategy proposed that employing a somewhat more intricate linear layer could yield improved balance between security and efficiency. We provide a more detailed explanation of the wide-trail strategy in Section 2.8.5.

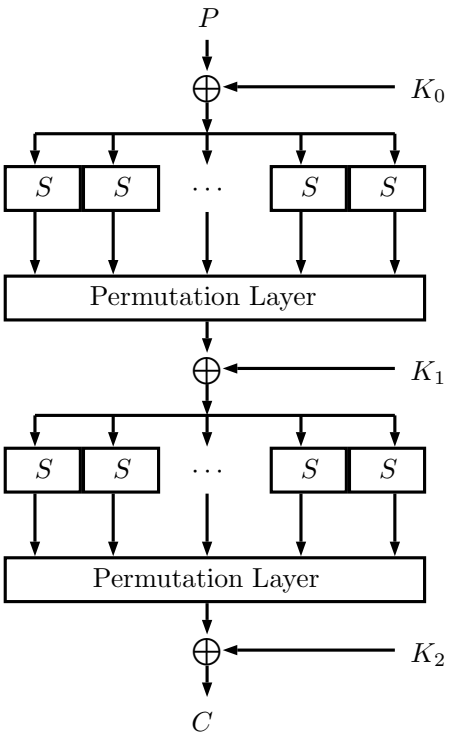


Figure 2-1: 2-round SPN

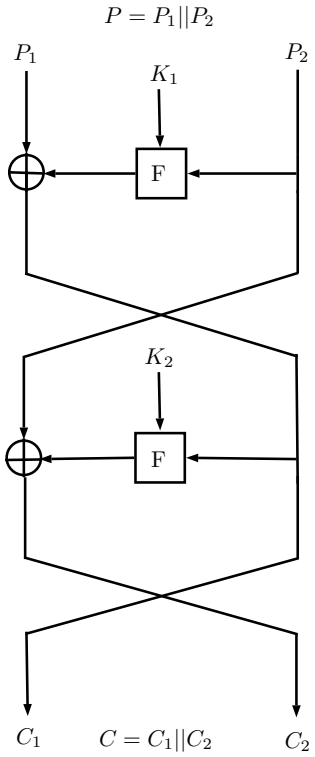


Figure 2-2: 2-round Feistel Network

2.8.2 Feistel Network

Feistel networks are named after Horst Feistel, a member of the IBM team that created the first commercial encryption standard, Data Encryption Standard (DES) [PUB77]. A notable feature of Feistel networks is that encryption and decryption are essentially the same, except for reversing the order of the round functions. This similarity is helpful in hardware implementations because it allows using the same circuit for both encryption and decryption. Figure 2-2 depicts two rounds of a Feistel network.

Its round function divides the internal state into two parts. It alternately uses one part as input for an F-function and inserts the F-function's output into the other

part. Unlike SPN, a key difference is that the F-function doesn't have to be invertible because the decryption process uses it in the same forward direction. However, the F-function's design can still follow SPN principles.

2.8.3 Security Notions

In the most severe scenario, a cryptanalyst aims to retrieve the user's secret key. However, an attacker might settle for less. With this consideration, it becomes feasible to create a hierarchy of potential attacks [Knu94].

(i) Total Break: The attacker recovers the user's secret key K.

(ii) Global Deduction: The attacker discovers an algorithm $\mathcal{A}$ that is functionally equivalent to either E or E^{-1}.

(iii) Local Deduction: The attacker has the ability to produce a message (or ciphertext) corresponding to a previously unseen ciphertext (or message).

(iv) Distinguishing Algorithm: The attacker can proficiently differentiate between two black boxes; one contains the block cipher with a randomly chosen encryption key, while the other contains a randomly chosen permutation.

We also need to consider the type of attackers we aim to safeguard against. Specifically, various assumptions can be made regarding the kind of data in the possession of the attackers.

(i) *Ciphertext-Only Attack (COA)*: The attacker can only observe the ciphertexts.

(ii) *Known-Plaintext Attack (KPA)*: The attacker can observe a limited number of plaintexts along with their corresponding ciphertexts.

(iii) *Chosen-plaintext attack (CPA)*: The attacker temporarily has access to the encryption algorithm, enabling them to choose the plaintexts and request their corresponding ciphertexts.

(iv) *Adaptive chosen-plaintext attack (CPA2)*: Apart from having temporary access to the encryption oracle, the attacker can analyze the responses from previous queries before selecting the next plaintext to query.

(v) *Chosen-ciphertext attack (CCA)*: The attacker temporarily has access to the decryption oracle, enabling them to select ciphertexts and inquire about their corresponding plaintexts.

(vi) *Adaptive chosen-ciphertext attack (CCA2)*: Besides having temporary access to the decryption oracle, the adversary can analyze the responses from previous queries before selecting the next ciphertext to query.

We have already provided some intuition on the requirements of a secure block cipher. Now, we will present a formal security definition for a block cipher.

Adversaries and Oracles: A cryptographic adversary $\mathcal{A}$ is a randomized algorithm with access to an oracle $\mathcal{O}$. The oracle $\mathcal{O}$ is itself an algorithm that provides cryptographic functionality or information for analysis and security assessment within a cryptographic scheme. The interaction between the adversary $\mathcal{A}$ and the oracle $\mathcal{O}$ results in a set of pairs $\{(x_1, y_1), (x_2, y_2), \ldots, (x_q, y_q)\}$. Here, $x_1, x_2, \ldots, x_q$ represent the q queries made by the adversary $\mathcal{A}$ to the oracle $\mathcal{O}$, and $y_1, y_2, \ldots, y_q$ are the corresponding responses from the oracle $\mathcal{O}$. The adversary is considered adaptive, meaning that the i-th query made by the adversary depends on the previous $i-1$ responses.

Distinguishing Advantage: We consider two systems S and R along with a distinguishing adversary $\mathcal{A}$. The adversary $\mathcal{A}$ is granted access to either S or R. Following the interaction with an oracle $\mathcal{O}$, $\mathcal{A}$ outputs 1, represented as $\mathcal{A}^{\mathcal{O}} \Rightarrow 1$. This type of adversary is referred to as a *distinguisher*, and the activity is termed a *distinguishing game*.

The objective of the distinguishing adversary, or distinguisher, is to differentiate between the two systems S and R within a distinguishing game. The distinguishing advantage of the distinguisher is defined as:

$$\mathbf{Adv}^{\mathsf{S}}_{\mathsf{R}}(\mathcal{A}) = \left| \ \Pr[\mathcal{A}^{\mathsf{S}} \Rightarrow 1] - \Pr[\mathcal{A}^{\mathsf{R}} \Rightarrow 1] \ \right|,$$

where the above probability refers to the probability computed over the probability spaces of the adversary $\mathcal{A}$ and the oracle $\mathcal{O}$. By considering the maximum advantages obtained from all potential distinguishers $\mathcal{A}$ making q queries, we define this maximum advantage as:

$$\max_{\mathcal{A}} \mathbf{Adv}^{\mathsf{S}}_{\mathsf{R}}(\mathcal{A}).$$

Adversary Resources: In the provided definition of the distinguishing advantage of the adversary $\mathcal{A}$, the resources employed by the distinguisher to differentiate

between algorithms S and R are not explicitly specified. Generally, adversaries commonly consider two primary resources: time complexity and query complexity.

The time complexity (t) of an adversary $\mathcal{A}$ encompasses the time needed for both interacting with the oracle and performing local computations. Query complexity (q) is defined as the count of queries made by $\mathcal{A}$ to the oracle. The maximum advantage in distinguishing S and R, considering a class of adversaries with a maximum time complexity of t and a maximum query complexity of q is defined as follows:

$$\mathbf{Adv}^{\mathsf{S}}_{\mathsf{R}}(q,t) = \max_{\mathcal{A}} \mathbf{Adv}^{\mathsf{S}}_{\mathsf{R}}(\mathcal{A}),$$

where the maximum is determined across all adversaries making up to q queries with a maximum running time of t.

Pseudorandom Permutation and Strong Pseudorandom Permutation: Let $E : \mathbb{F}_2^n \times \mathbb{F}_2^k \mapsto \mathbb{F}_2^n$ be a block cipher, which is denoted by $E(P, K)$ or $E_K(P)$. For each key $K \in \mathbb{F}_2^k$, the map $E_K()$ is a permutation over the domain space $\mathbb{F}_2^n$. Now consider a distinguisher $\mathcal{A}$, who has oracle access to either E_K where K is chosen uniformly from $\mathbb{F}_2^k$ or a permutation chosen uniformly from $\mathrm{Perm}(\mathbb{F}_2^n)$, where $\mathrm{Perm}(\mathbb{F}_2^n)$ denotes the set of all permutations over $\mathbb{F}_2^n$. Suppose $\mathcal{A}$ makes at most q queries and runs for the time at most t.

The task of $\mathcal{A}$ is to distinguish E_K from a random permutation. We consider the *pseudorandom permutation (PRP) advantage of* $\mathcal{A}$ as

$$\mathbf{Adv}^{\mathrm{PRP}}_{E}(\mathcal{A}) = \mid \Pr[\mathcal{A}^{E_K} \Rightarrow 1] - \Pr[\mathcal{A}^{\rho} \Rightarrow 1] \mid,$$

where $K \xleftarrow{\$} \mathbb{F}_2^k$ and $\rho \xleftarrow{\$} \mathrm{Perm}(\mathbb{F}_2^n)$. E is said to be a (q, t, ϵ) *secure PRP*, if

$$\mathbf{Adv}^{\mathsf{PRP}}_{E}(q,t) = \max_{\mathcal{A}} \mathbf{Adv}^{\mathsf{PRP}}_{E}(\mathcal{A}) \leq \epsilon,$$

where the maximum is taken over all adversaries with maximum running time t that asks at most q queries.

Now, we define the security against those adversaries who have access to the block cipher as well as their inverse. Consider a distinguisher $\mathcal{A}$, who has oracle access to a permutation and its inverse over $\mathbb{F}_2^n$. Suppose $\mathcal{A}$ makes at most q queries with maximum running time t. The task of the distinguisher is to distinguish E_K from a random permutation. We consider the *strong pseudorandom permutation (SPRP)*

advantage of $\mathcal{A}$ as

$$\mathbf{Adv}_E^{\mathrm{SPRP}}(\mathcal{A}) = |\ \Pr[\mathcal{A}^{E_K, E_K^{-1}} \Rightarrow 1] - \Pr[\mathcal{A}^{\rho, \rho^{-1}} \Rightarrow 1]\ |,$$

where $K \xleftarrow{\$} \mathbb{F}_2^k$ and $\rho \xleftarrow{\$} \mathrm{Perm}(\mathbb{F}_2^n)$. E is said to be a (q, t, ϵ) *secure SPRP*, if

$$\mathbf{Adv}_E^{\mathsf{SPRP}}(q, t) = \max_{\mathcal{A}} \mathbf{Adv}_E^{\mathsf{SPRP}}(\mathcal{A}) \leq \epsilon,$$

where the maximum is taken over all adversaries that make at most q queries with maximum running time t.

The concepts of a pseudorandom permutation and a strong pseudorandom permutation formalize the idea of security against chosen-plaintext and chosen-ciphertext attacks, respectively. However, these concepts can only define security up to a certain threshold (ϵ) and solely against adversaries with limited resources (q, t). Consequently, in practical scenarios, it is necessary to assess realistic assumptions about the adversary and determine the specific level of security one aims to achieve.

2.8.4 Classical Cryptanalysis Techniques

Here, we provide a brief introduction to the two most widely employed cryptanalysis techniques for block ciphers- *differential cryptanalysis* and *linear cryptanalysis*.

Differential Cryptanalysis: Biham and Shamir introduced the technique of *differential cryptanalysis* in 1990 [BS91a, BS91b] as an attack on DES [PUB77]. Here, we primarily outline the most important concepts from the literature. For a comprehensive study, we refer to [Bei18, DR02, Eic18, Hey02].

Differential cryptanalysis is a chosen-plaintext attack that exploits a differential $\alpha \xrightarrow{E_K} \beta$ over E_K with high probability i.e. $\Pr[\alpha \xrightarrow{E_K} \beta] > 2^{-n}$. The adversary can differentiate E_K from a random permutation by making multiple queries to the oracle $\mathcal{O}$ with randomly chosen input pairs $(x, x \oplus \alpha)$ and verifying if the output difference $\mathcal{O}(x) \oplus \mathcal{O}(x \oplus \alpha)$ matches β as frequently as expected based on the given differential probability.

Since the block cipher is a set of permutations determined by a key, it is necessary to distinguish between the *fixed-key probability of a differential* (α, β) and the *expected differential probability* when averaged across all possible keys. Formally, for a block cipher $E\colon \mathbb{F}_2^n \times \mathbb{F}_2^k \mapsto \mathbb{F}_2^n$, the *expected differential probability* of a differential (α, β) is

defined as follows:

$$\mathrm{EDP}_E(\alpha, \beta) = \frac{1}{2^k} \sum_{K \in \mathbb{F}_2^k} \Pr[\alpha \xrightarrow{E_K} \beta].$$

The next three paragraphs delve into the concept of a *differential characteristic* and the estimation of the probability of a differential characteristic in a practical scenario. It is noteworthy that the content in these paragraphs has been primarily influenced by Section 1 and Section 2.1 of the paper [BR22].

For functions expressed as a composition of simple operations, the conventional approach involves examining sequences of intermediate differences or *characteristics*. The probability of a characteristic is then estimated heuristically by multiplying the probabilities of the intermediate differentials. In the realm of block ciphers, Lai, Massey, and Murphy [LMM91] demonstrated that this method accurately calculates the key-averaged probability for Markov ciphers.

However, during a differential attack, as the key remains constant, computing the average data-complexity becomes challenging solely based on the average probability of differentials . Therefore, Lai et al. [LMM91] introduced an additional assumption known as the *Hypothesis of Stochastic Equivalence*.

Hypothesis of Stochastic Equivalence: Given a block cipher E and a differential (α, β),

$$\mathrm{EDP}_E(\alpha, \beta) \approx \Pr[\alpha \xrightarrow{E_K} \beta],$$

for almost all keys K.

In practice, it turns out that the probability can vary significantly between keys. Hence, standard assumptions may lead to incorrect conclusions. Furthermore, averages may hide weak key attacks that can considerably degrade security. Finally, the same formalism is used even when there is no key, such as for cryptographic permutations, or when the cryptanalyst has full control over the key, such as in many hash functions.

Differential cryptanalysis is commonly employed on functions $F\colon \mathbb{F}_2^n \mapsto \mathbb{F}_2^n$ structured as $F = F_r \circ \cdots \circ F_1$, where the individual functions F_i exhibit differentials with relatively high probabilities and are usually easier to analyze. In such instances, the probability of a differential (a_1, a_{r+1}) can be approximated using characteristics. A *differential characteristic* (also known as differential trail) is a sequence $(a_1, a_2, \ldots, a_{r+1})$ consisting of compatible intermediate input and output differences for each of the functions F_i. It holds that

$$\Pr[F(x) \oplus F(x \oplus a_1) = a_{r+1}] = \sum_{a_2,\ldots,a_r \in \mathbb{F}_2^n} \Pr[\wedge_{i=1}^{r} F_i(x_i) \oplus F_i(x_i \oplus a_i) = a_{i+1}],$$

with x_1 uniformly random on $\mathbb{F}_2^n$ and $x_i = F_{i-1}(x_{i-1})$ for $i = 2, \ldots, r$. The probability of a characteristic is often estimated using the assumption that intermediate differentials are independent. Thus,

$$\Pr[\wedge_{i=1}^{r} F_i(x_i) \oplus F_i(x_i \oplus a_i) = a_{i+1}] = \prod_{i=1}^{r} \Pr[F_i(x_i) \oplus F_i(x_i \oplus a_i) = a_{i+1}].$$

Therefore, the expected differential probability of a differential (a_1, a_{r+1}) can be given as

$$\mathrm{EDP}_E(a_1, a_{r+1}) \approx \sum_{a_2,\ldots,a_r \in \mathbb{F}_2^n} \prod_{i=1}^{r} \Pr[F_i(x_i) \oplus F_i(x_i \oplus a_i) = a_{i+1}]. \tag{2.3}$$

In the context of differential cryptanalysis, the Sboxes that are involved in a characteristic and exhibit a nonzero input difference (and consequently a nonzero output difference) are called *active Sboxes*. Generally, if the active Sboxes have higher differential probabilities, the overall characteristic probability for the entire cipher is larger. Also, a larger characteristic probability is achieved when there are fewer active Sboxes. The estimated number of plaintext pairs required for a differential attack is $N_D = cp_D^{-1}$ [Hey02], where c is a small constant, and p_D is the differential characteristic probability for the $r-1$ rounds of the r-round cipher.

To mount a differential cryptanalysis on an n-bit block cipher, there needs to be a differential probability higher than 2^{-n}. A common approach to assess the resistance of a block cipher to differential cryptanalysis is to determine a lower bound for the number of active Sboxes, N_S, in any r-round differential characteristic. The differential probability of any r-round differential characteristic is upper bounded by $(\Delta_S)^{N_s}$, where Δ_S is maximum differential probability of the Sbox in the cipher. Therefore, if a substantially reduced-round version with $(\Delta_S)^{N_s} < 2^{-n}$, it is considered that the cipher is resistant to differential cryptanalysis.

Linear Cryptanalysis: In 1993, Matsui [Mat94] introduced the *linear cryptanalysis* as a new known-plaintext attack on DES. It is based on probabilistic linear relations or *linear approximations*, a concept that was first used in [TCG92]. Here,

we primarily outline the most important concepts from the literature. For a comprehensive study, we refer to [Dae95, DR02, Hey02].

According to [Mat94], around $c{\cdot}|\mathrm{Cor}(\alpha \xrightarrow{E_K} \beta)|^2$ known plaintexts are necessary for a distinguisher of E_K to have a reasonably high advantage, where c is a small constant. So, if the absolute correlation is less than $2^{-n/2}$, there would not be sufficient plaintexts available. In other words, to mount linear cryptanalysis on E_K an adversary must have a linear approximation (α, β) for E_K, which has a high absolute correlation, i.e., $|\mathrm{Cor}(\alpha \xrightarrow{E_K} \beta)| > 2^{-n/2}$. The adversary can now distinguish E_K from a random permutation by queries to the oracle $\mathcal{O}$ about various inputs x and verifying if $x \cdot \alpha \oplus \mathcal{O}(x) \cdot \beta = 0$ holds as frequently as expected based on the correlation of the linear approximation.

A linear approximation (a_1, a_{r+1}) for an iterative function $E_K = R_K^{(r)} \circ \cdots \circ R_K^{(2)} \circ R_K^{(1)}$, can be analyzed using *linear trails*. A linear trail is a sequence $(a_1, a_2, \ldots, a_{r+1})$ of compatible intermediate input and output masks for each of the functions $R_K^{(i)}$. The correlation of the linear approximation (a_1, a_{r+1}) can be expressed as the sum of the correlations of all the linear trails it contains [DGV95] i.e.,

$$\mathrm{Cor}(a_1 \xrightarrow{E_K} a_{r+1}) = \sum_{a_2,\ldots,a_r \in \mathbb{F}_2^n} \prod_{i=1}^{r} \mathrm{Cor}(a_i \xrightarrow{R_K^{(i)}} a_{i+1}).$$

If the cipher operates as a key-alternating cipher [2] [DR02] with independent round keys, the correlation of the keyed round function can be described in relation to the correlation of the unkeyed round function:

$$\mathrm{Cor}(a_1 \xrightarrow{E_K} a_{r+1}) = \sum_{a_2,\ldots,a_r \in \mathbb{F}_2^n} (-1)^{\oplus_{i=1}^{r} K_i \cdot a_i} \prod_{i=1}^{r} \mathrm{Cor}(a_i \xrightarrow{R^{(i)}} a_{i+1}),$$

where $(K_1, \ldots, K_r)$ denote the round keys that are derived from K by the key scheduling algorithm.

In [Nyb95], Nyberg introduced the concept of the expected linear potential as the average value of the squared correlation for a randomly chosen key. It is also demonstrated that this expected linear potential is equal to the sum of the squared correlations over all linear trails, under the assumption of a key-alternating cipher with independent keys:

[2] This type of cipher exactly describes the way the particular round keys are introduced within the rounds. The key-scheduling function has to generate the round key $K_i \in \mathbb{F}_2^n$ for each round function R_i and the round function R_i is defined as $R_i \colon \mathbb{F}_2^n \times \mathbb{F}_2^n \mapsto \mathbb{F}_2^n$.

$$\begin{aligned}\mathrm{ELP}(a_1 \xrightarrow{E_K} a_{r+1}) &= \mathrm{Exp}_K(\mathrm{Cor}(a_1 \xrightarrow{E_K} a_{r+1})^2) \\ &= \sum_{a_2,\ldots,a_r \in \mathbb{F}_2^n} \prod_{i=1}^{r} \mathrm{Cor}(a_i \xrightarrow{R^{(i)}} a_{i+1})^2.\end{aligned}$$

Similar to the rationale for resistance against differential cryptanalysis, the standard argument for a designer ensuring resistance against linear cryptanalysis relies on a single linear trail. The objective is specifically to secure a low upper bound (i.e., $< 2^{-n/2}$) on the absolute correlation of any nonzero linear trail over a reduced-round version of the cipher. The bound can be determined by the number of *active Sboxes*, N_S, in any r-round linear trail. An Sbox of a specific round is said to be active with respect to a linear trail if its output selection vector is nonzero for that linear trail. By the Piling Up Lemma, the bias of a linear approximation is upper bounded by $2^{N_S-1}(\mathcal{E}_S)^{N_s}$, where $\mathcal{E}_S$ is the maximal absolute bias of a linear approximation of the Sbox in the cipher. Thus, the absolute correlation of any r round is upper bounded by $2^{N_S}(\mathcal{E}_S)^{N_s}$. Therefore, if a substantially reduced-round version with $(2\mathcal{E}_S)^{N_s} < 2^{-n/2}$, it is considered that the cipher is resistant to linear cryptanalysis [3].

2.8.5 The Wide-Trail Strategy and AES-like Ciphers:

In this section, we briefly discuss the *wide-trail strategy* introduced by Daemen in [Dae95]. For a more detailed overview, we refer to [Dae95, DR02]. It suggests a design approach for key-alternating block ciphers that facilitates simple arguments about their resistance to differential and linear attacks. The main starting point is to express the round functions R_i of the cipher as $R_i = L \circ \mathcal{N}$, where $\mathcal{N}$ is an Sbox layer composed of parallel applications of b-bit Sbox S, and L is a linear layer represented by $M \in GL(n, \mathbb{F}_2)$ as $x \mapsto Mx$. Instead of using a bit permutation for L, the wide-trail strategy explains how the linear layer could be chosen in a more general way to avoid the existence of differential (or linear) trails with high probability (or absolute correlation).

In [Dae95], Daemen introduced the concept of the branch number of a linear transformation as a measure of its diffusion. Essentially, it provides the minimum number of active Sboxes we can expect in any valid differential (or linear) trail over two rounds. Specifically, for any two-round differential (or linear) trail, the number of active Sboxes is lower bounded by $\beta_d(L)$ (or $\beta_l(L)$).

[3] $\mathrm{Cor}_S = 2\mathcal{E}_S$ is called the *maximum absolute correlation* of a linear approximation of the Sbox S. Thus, the absolute correlation of any r round is upper bounded by $(\mathrm{Cor}_S)^{N_S}$.

AES-like Ciphers: Now, we will describe a block cipher structure specifically designed using the wide-trail strategy. Originally, the structure was introduced with the block cipher SQUARE [DKR97], a precursor to the Rijndael [DR99] cipher adopted as the Advanced Encryption Standard (AES) in 2001 [PUB01]. Since AES has inspired many other designs, such as ANUBIS [BR00a], LED [GPPR11], Midori [BBI+15], Prince [BCG+12], QARMA [Ava17], Skinny and Mantis [BJK+16] let us start with a more general definition. We refer to ciphers designed based on this general notion as *AES-like ciphers* [Bei18].

An AES-like cipher aligns with the concept of an SPN cipher. Specifically, it functions as a key-alternating block cipher with a block length of $n = b \cdot n_b$, where n_b is further factored into two positive integers n_r and n_c, i.e., $n_b = n_r \cdot n_c$. To enhance representation, the cipher's input, output, and internal states $\mathbb{F}_2^{bn_rn_c}$ are typically expressed as an $n_r \times n_c$-dimensional array with b-bit words. Thus,

$$x = \begin{bmatrix} x_1 & x_{n_r+1} & \dots & x_{(n_c-1)n_r+1} \\ x_2 & x_{n_r+2} & \dots & x_{(n_c-1)n_r+2} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n_r} & x_{2n_r} & \dots & x_{n_cn_r} \end{bmatrix}, \text{ where } x_i \in \mathbb{F}_2^b.$$

Note that since $\mathbb{F}_{2^b} \cong \mathbb{F}_2^b$ as a vector space over $\mathbb{F}_2$, we can represent a b-bit word x_i as an element of the finite field with 2^b elements.

An AES-like cipher is defined by using a specific type of round function, as described in Definition 2.9. After applying this round function (unkeyed), a round key $K^{(r)} \in \mathbb{F}_2^{bn_rn_c}$, is added to the internal state of the cipher. This addition of a round key is a common feature in key-alternating ciphers. Importantly, the concept of an AES-like cipher does not specify any particular requirements for the key-scheduling algorithm. For simplicity, we will not consider it in the following discussions.

Definition 2.47. *[Bei18] An AES-like round is defined as a permutation*

$$R_{S,\rho,M} \colon \mathbb{F}_{2^b}^{n_rn_c} \to \mathbb{F}_{2^b}^{n_rn_c},$$

which is parametrized by the word length b, the state dimensions $n_r \times n_c$, a b-bit Sbox S, a permutation $\rho \in \mathcal{S}_{n_rn_c}$ and $M \in GL(n_r, \mathbb{F}_{2^b})$. In particular, the round function $R_{S,\rho,M}$ is composed of the bijective transformations $SubCell_S$, $Permute_\rho$ and $MixColumn_M$ operating on an $n_r \times n_c$ state, such that $R_{S,\rho,M} = MixColumn_M \circ Permute_\rho \circ SubCell_S$:

(i) *$SubCell_S$ is a parallel application of the Sbox S to all $n_r \times n_c$ words of the state.*

$$SubCell_S \colon \mathbb{F}_{2^b}^{n_r n_c} \to \mathbb{F}_{2^b}^{n_r n_c}$$
$$\text{for } 0 \le i < n_r,\ 0 \le j < n_c \colon\ x_{n_r j+i+1} \mapsto S(x_{n_r j+i+1}).$$

(ii) *$Permute_\rho$ permutes the permutation according to the permutation ρ, i.e.,*

$$Permute_\rho \colon \mathbb{F}_{2^b}^{n_r n_c} \to \mathbb{F}_{2^b}^{n_r n_c}$$
$$\text{for } 0 \le i < n_r,\ 0 \le j < n_c \colon\ x_{n_r j+i+1} \mapsto x_{\rho(n_r j+i+1)}.$$

(iii) *$MixColumn_M$ applies a multiplication by the $n_r \times n_r$ matrix M to all columns of the state, i.e.,*

$$MixColumn_M \colon \mathbb{F}_{2^b}^{n_r n_c} \to \mathbb{F}_{2^b}^{n_r n_c}$$
$$\text{for } 0 \le j < n_c \colon\ [x_{n_r j+1}, \ldots, x_{n_r j+n_r}]^T \mapsto M \cdot [x_{n_r j+1}, \ldots, x_{n_r j+n_r}]^T.$$

The AES-like design has a simple structure, and if the permutation ρ is chosen carefully, a strong lower bound on the number of active Sboxes of any valid four-round trail can be proven. Specifically, the minimum number of active Sboxes in any valid four-round trail is guaranteed to be at least the square of the branch number of the linear transformation $x \mapsto Mx$.

Theorem 2.11. *[Bei18, DR02] Let $R_{S,\rho,M}$ be an AES-like round with $n_c \ge n_r$ such that, for each column of the state, $Permute_\rho$ distributes the word of a column to all different columns. Then, the minimum number of active Sboxes of any four-round differential (or linear) trail is lower bounded by $\beta_d(M)^2$ (or $\beta_l(M)^2$).*

Since an MDS matrix M of order n has branch number $\beta_d(M) = \beta_l(M) = n + 1$, the above theorem asserts that in an AES-like cipher with a round function $R_{S,\rho,M}$, if the matrix M is an MDS matrix, then the minimum number of active Sboxes in any four-round differential (or linear) trail is lower bounded by $(n+1)^2$. In AES [DR02], we know that M is an MDS matrix of order 4. Consequently, the minimum number of active Sboxes in any four-round differential (or linear) trail is lower bounded by 25. Since the Sbox in AES has a maximum differential probability (or maximum absolute correlation) of 2^{-6} (or 2^{-3}), the differential probability (or absolute correlation) of any differential (or linear) trail is upper bounded by 2^{-150} (or 2^{-75}).

3

MDS Matrix Construction over Finite Fields: A Comprehensive Study of Various Matrix Structures

Contents

3.1 Introduction

The optimal branch number of MDS matrices has made them a preferred choice for designing diffusion layers in many block ciphers and hash functions. As a result, several methods have been proposed for designing MDS matrices. There are two main approaches to constructing MDS matrices: nonrecursive and recursive. In recursive constructions, a sparse matrix B of order n is generally chosen, and the elements of the matrix are selected in such a way that B^n becomes an MDS matrix. In

nonrecursive constructions, the constructed matrices themselves are MDS. Another way to classify the techniques used to find MDS matrices is based on whether the matrix is constructed directly or a search method is employed by enumerating a search space. Direct constructions use algebraic properties to provide an MDS matrix, while in search methods, elements of the matrix are judiciously selected, and it is checked whether the matrix is MDS or not. It should be noted that the problem of verifying whether a matrix is MDS or not is NP-complete. Therefore, the search technique is useful only for finding MDS matrices of small orders.

There are two main direct methods for constructing nonrecursive MDS matrices: one is from a Cauchy matrix and the other is from two Vandermonde matrices. These methods provide MDS matrices of any order, but these matrices are generally not efficient for implementation. So, we use the search method in nonrecursive constructions that output efficiently implementable MDS matrices.

One popular technique for such constructions is to search for elements of a circulant matrix. It is worth noting that the circulant MDS matrix used in AES has been found using a search method. There are several circulant-like matrices [GR15] and generalized circulant matrices [LS16] which are also used in such constructions. Toeplitz matrices and Hankel matrices are deeply interconnected with circulant matrices. Recently, Toeplitz matrices have been used to construct MDS matrices [SS16, SS17]. Similar to circulant, circulant-like and generalized circulant MDS matrices, search methods are used to construct Toeplitz MDS matrices.

As in nonrecursive constructions, there are several direct recursive constructions as well. However, as before, they are not so efficient for implementation and search methods provide efficient MDS matrices of low order.

In the general context of implementation of block ciphers, we note that if an efficient MDS matrix M used in encryption, happens to be involutory or orthogonal, then its inverse M^{-1} applied for decryption will also be efficient. So, it is of special interest to find efficient MDS matrices that are also involutory or orthogonal.

In this chapter, we provide a brief survey on MDS matrices. In addition to providing a summary of existing results, we make several contributions. We exhibit some deep and nontrivial interconnections between different constructions of MDS matrices. For example, we prove that all known Vandermonde constructions are basically equivalent to Cauchy constructions. We prove some folklore results which are used in MDS matrix literature. Wherever possible, we provide some simpler alternative proofs. We do not discuss efficiency issues or hardware implementations; however, the theory accumulated and discussed here should provide an easy guide towards

efficient implementations. We find a gap in one of the lemmas in the paper [AF15, Lemma 1] and then provide the correct statement in Subsection 3.7.2. The result is stated in Lemma 3.26 followed by an example that shows the existence of a gap in the statement of Lemma 1 of the paper [AF15].

Outline: The rest of this chapter is structured as follows: In Section 3.2, we discuss various constructions of MDS matrices from Cauchy matrices. Section 3.3 covers the various constructions of MDS matrices from Vandermonde matrices. The interconnection and equivalence between these two constructions are highlighted in Section 3.4. To overcome the inefficiencies of direct constructions, we move on to constructions using the search method in Sections 3.5 and 3.6. Section 3.5 discusses the constructions of MDS matrices from circulant matrices and their variants while Section 3.6 covers the constructions of MDS matrices from Toeplitz and Hankel matrices. In Section 3.7, we describe direct constructions of recursive MDS matrices. Finally, the chapter concludes in Section 3.8.

Other Notations: In this chapter, we assume that the row and column indices of a matrix start from 0 for the purpose of simplifying the proofs of some results. Thus, for an $n \times n$ matrix A, $(A)_{i,j}$ denotes the element at the (i, j)-th position of the matrix, where $0 \leq i, j \leq n-1$.

3.2 Constructing MDS Matrices from Cauchy Matrices

Application of Cauchy matrices for constructing MDS codes are widely available in literature [CJK15, GR13a, MS77, MRS12, RS85, RL89, SKOP15, YMT97]. Youssef et al. used Cauchy matrix for constructing MDS matrices with efficient cryptographic applications in mind [YMT97]. Gupta et al. [GR13a] used similar methods in a more formal setup. Cui et al. [CJK15] define compact Cauchy matrix and provide several interesting results. Mattoussi et al. [MRS12] used triangular array to construct MDS codes, which is related with Cauchy matrices [MRS12, RS85]. We will mainly discuss [CJK15, GR13a, MRS12, RS85, RL89, YMT97] in this section.

From Fact 2.8, we know that Cauchy matrices are nonsingular and from Fact 2.9, we know that any square submatrix of a Cauchy matrix is again a Cauchy matrix. This property enables us to construct MDS matrices and the following lemma summarizes these observations.

Lemma 3.1. *[GR13a, Lemma 1][MS77, Page 323][YMT97] Let $x_0, x_1, \ldots, x_{n-1}$ and $y_0, y_1, \ldots, y_{n-1}$ be distinct elements such that $x_i + y_j \neq 0$ for all $0 \leq i, j \leq n-1$. Then, the matrix $A = (a_{i,j})$, where $a_{i,j} = \frac{1}{x_i + y_j}$, forms an MDS matrix.*

We will call this construction as Cauchy based construction of type 1. Depending on the nature of x_i's and y_i's there are basically four types of constructions available in the literature. We will call this type 1, type 2, type 3 and type 4 constructions and we will come back to it whenever we discuss them.

Remark 3.1. *One special case of Lemma 3.1 is that y_i is of the form $l + x_i$, where l is an arbitrary nonzero element in $\mathbb{F}_{2^r}$. We will call this construction as Cauchy based construction of type 2.*

The following lemma and its corollary studies the number of distinct entries in the construction using Lemma 3.1 which is crucial for studying the construction of efficient MDS matrices from Cauchy matrices.

Lemma 3.2. *[GR13a, Lemma 2] In the $n \times n$ MDS matrix A constructed using the method in Lemma 3.1, each row (or column) consists of n distinct elements.*

Proof. For the i-th row of A, if possible, let $(A)_{i,j_1} = (A)_{i,j_2}$, for any two $j_1, j_2 \in \{0, \ldots, n-1\}$ such that $j_1 \neq j_2$. Then, we have $\frac{1}{x_i + y_{j_1}} = \frac{1}{x_i + y_{j_2}}$ which implies that $y_{j_1} = y_{j_2}$. This observation contradicts the assumption that the elements y_j are distinct. Since the choice of i was arbitrary, this result applies to all rows of matrix A. The proof for the columns follows a similar argument. □

Since each row (or column) of the matrix constructed by Lemma 3.1 has n distinct elements, we have the following corollary.

Corollary 3.1. *[GR13a, Corollary 1] The $n \times n$ MDS matrix A, constructed according to Lemma 3.1, possesses a minimum of n distinct elements.*

Example 3.1. *Let α be the primitive element of $\mathbb{F}_{2^4}$ whose constructing polynomial is $x^4 + x + 1$. Let $x_0 = 0$, $x_1 = \alpha^4$, $x_2 = \alpha^8$ and $y_0 = 1$, $y_1 = \alpha^3$, $y_2 = \alpha^5$. Then the matrix A using the Lemma 3.1 is given by A*

$$= \begin{bmatrix} 1 & \frac{1}{\alpha^3} & \frac{1}{\alpha^5} \\ \frac{1}{(1+\alpha^4)} & \frac{1}{(\alpha^3+\alpha^4)} & \frac{1}{(\alpha^4+\alpha^5)} \\ \frac{1}{(1+\alpha^8)} & \frac{1}{(\alpha^3+\alpha^8)} & \frac{1}{(\alpha^4+\alpha^8)} \end{bmatrix} = \begin{bmatrix} 1 & \alpha^3+\alpha^2+\alpha+1 & \alpha^2+\alpha+1 \\ \alpha^3+1 & \alpha^2+1 & \alpha^3+\alpha+1 \\ \alpha^3+\alpha^2+1 & \alpha^2 & \alpha^3+\alpha^2+\alpha \end{bmatrix}$$

is MDS but not involutory. Note that each row (and column) has $n = 3$ distinct elements and total number of distinct element is 9.

The following is an example of the special case of Lemma 3.1.

Example 3.2. *Let* $x_0 = \alpha,\ x_1 = \alpha^2,\ x_2 = \alpha^3$ *and* $y_i = l + x_i$ *for* $0 \leq i \leq 2$, *where* $l = 1$. *Therefore,*

$$A = \begin{bmatrix} 1 & \frac{1}{(1+\alpha+\alpha^2)} & \frac{1}{(1+\alpha+\alpha^3)} \\ \frac{1}{(1+\alpha+\alpha^2)} & 1 & \frac{1}{(1+\alpha^2+\alpha^3)} \\ \frac{1}{(1+\alpha+\alpha^3)} & \frac{1}{(1+\alpha^2+\alpha^3)} & 1 \end{bmatrix} = \begin{bmatrix} 1 & \alpha^2+\alpha & \alpha^2+1 \\ \alpha^2+\alpha & 1 & \alpha^2 \\ \alpha^2+1 & \alpha^2 & 1 \end{bmatrix}$$

is MDS but not involutory. Note that here each row (and column) has $n = 3$ *distinct elements and total number of distinct elements is 4.*

According to Corollary 3.1, a square matrix of order n constructed using Lemma 3.1 will have a minimum of n distinct elements. In [GR13a], the authors constructed MDS matrices of order n that contain exactly n distinct elements. This approach offers a twofold advantage. Firstly, it allows for the selection of only n suitable and efficient elements, which can be chosen based on low implementation cost. Secondly, this enables the construction of MDS matrices using the Cauchy construction method. It is worth noting that, in order to construct efficient MDS matrices, it is advantageous to minimize the number of distinct elements to reduce the implementation overheads [JV05b].

In [CJK15] authors called such MDS Cauchy matrices having exactly n elements as compact Cauchy matrices. Formally, let an $n \times n$ matrix $A_X = (a_{i,j})$ be a Cauchy matrix generated by the vector $X = (x_0, x_1, \ldots, x_{n-1}, x_n, \ldots, x_{2n-1})$ i.e. $a_{i,j} = \frac{1}{x_i + x_{n+j}}$. Then A_X is called a compact Cauchy matrix if A_X precisely has n distinct entries.

Remark 3.2. *We will call an MDS matrix* A *of order* n *as compact MDS matrix if the number of distinct elements in* A *is* $\leq n$.

Lemma 3.3. *[GR13a, Lemma 6] Let* $G = \{x_0, x_1, \ldots, x_{n-1}\}$ *be an additive subgroup of* $\mathbb{F}_{2^r}$. *Consider the coset* $l+G$, *where* $l \notin G$, *with elements* $y_j = l + x_j$, *where* $x_j \in G$ *and* $0 \leq j \leq n-1$. *Then the* $n \times n$ *matrix* $A = (a_{i,j})$, *where* $a_{i,j} = \frac{1}{x_i + y_j}$, *for all* $0 \leq i, j \leq n-1$ *is an MDS matrix.*

Proof. To establish the proof, we first demonstrate that $x_i + y_j \neq 0$ holds for all $0 \leq i, j \leq n-1$. Consider $x_i + y_j = x_i + l + x_j = l + x_i + x_j \in l + G$. However, since $0 \notin l + G$ (due to $l \notin G$ and $0 \in G$), it follows that $x_i + y_j \neq 0$ for all $0 \leq i, j \leq n-1$. Also, all x_i are distinct elements of the group G, while y_j are distinct elements of the coset $l + G$. Consequently, based on Lemma 3.1, we can conclude that the matrix A is an MDS matrix. □

We will call this construction as Cauchy based construction of type 3.

Remark 3.3. *Lemma 3.3 provides the construction of MDS matrices of order n when n is a power of 2. However, when n is not a power of 2 and $n < 2^{r-1}$, the construction of an $n \times n$ MDS matrix over $\mathbb{F}_{2^r}$ is carried out in two steps. Firstly, we utilize Lemma 3.3 to construct an $2^m \times 2^m$ MDS matrix A' over $\mathbb{F}_{2^r}$, where $2^{m-1} < n < 2^m$. Subsequently, in the second step, we select an $n \times n$ submatrix A of our preference from A' by choosing n rows and n columns.*

Remark 3.4. *Lemma 3.3 is a particular case of Lemma 3.1.*

Lemma 3.4. *[GR13a, Lemma 7] The $n \times n$ matrix A mentioned in Lemma 3.3 is characterized by having precisely n distinct entries.*

Proof. The i-th row of the matrix has elements given by $a_{i,j} = \frac{1}{l+x_i+x_j}$ for $j = 0, 1, \ldots, n-1$. It is worth noting that the elements x_j form an additive group G, and when $x_i + x_j$ is computed for $j = 0, 1, \ldots, n-1$ with a fixed i, it produces all n distinct elements of G. Consequently, $l + x_i + x_j$ for $j = 0, 1, \ldots, n-1$ generates all n distinct elements of $l + G$. As i is arbitrary, each row of matrix A contains n distinct elements. Also, these elements represent the multiplicative inverses of the elements in $l + G$ within the field $\mathbb{F}_{2^r}$. Thus, the matrix A consists of precisely n distinct elements. □

Corollary 3.2. *[GR13a, Corollary 3] The matrix A of Lemma 3.3 is symmetric and all rows are the permutations of the first row.*

Proof. $a_{i,j} = a_{j,i} = \frac{1}{x_i+y_j} = \frac{1}{l+x_i+x_j}$ for all $0 \leq i, j \leq n-1$. Therefore, A is symmetric. The second part is directly follows from Lemma 3.2 and Lemma 3.4. □

In [GR13a], authors provided a sufficient condition (Lemma 3.3 of this chapter) for a Cauchy MDS matrix to be a compact Cauchy but did not discuss about the converse part. Later in [CJK15], authors provided a necessary and sufficient condition for the Cauchy matrix of order n to have exactly n distinct elements.

Theorem 3.1. *[CJK15, Theorem 1] A_X is an $n \times n$ compact Cauchy matrix over $\mathbb{F}_{2^r}$ generated by a vector $X = (x_0, \ldots, x_{n-1}, x_n, \ldots, x_{2n-1})$ if and only if there exists a additive subgroup H of $\mathbb{F}_{2^r}$ and $a, b \in \mathbb{F}_{2^r}$ such that $a+b \notin H$, $a+H = \{x_0, x_1, \ldots, x_{n-1}\}$, $b + H = \{x_n, x_{n+1}, \ldots, x_{2n-1}\}$.*

Proof. If $A_X = (a_{i,j})$ is a compact Cauchy matrix, then for all $i \in \mathbb{Z}_n$, we have

$$\{a_{i,0}, a_{i,1}, \ldots, a_{i,n-1}\} = \{a_{0,0}, a_{0,1}, \ldots, a_{0,n-1}\}.$$

Since the set $\{a_{i,0}, a_{i,1}, \ldots, a_{i,n-1}\}$ contains distinct entries, we may define a permutation $\pi_i : \mathbb{Z}_n \to \mathbb{Z}_n$ such that $a^{-1}_{i,\pi_i(t)} = a^{-1}_{0,t} = a^{-1}_{j,\pi_j(t)}$.

Note that $a^{-1}_{i_1,i_2} = x_{i_1} + x_{n+i_2}$ for all $i_1, i_2 \in \mathbb{Z}_n$. We have for any $i, j, t \in \mathbb{Z}_n$

$$a^{-1}_{i,\pi_i(t)} = x_i + x_{n+\pi_i(t)} = x_0 + x_{n+t} = x_j + x_{n+\pi_j(t)} = a^{-1}_{j,\pi_j(t)}. \tag{3.1}$$

Moreover, if $i \neq j$, then $x_{n+\pi_i(t)} + x_{n+\pi_j(t)} = x_i + x_j \neq 0$ (from Equation 3.1) which is followed by $\pi_i(t) \neq \pi_j(t)$. Hence, for any $t \in \mathbb{Z}_n$, it holds $\{\pi_i(t) : i \in \mathbb{Z}_n\} = \mathbb{Z}_n$. In other words,

$$\{(k, s) : k, s \in \mathbb{Z}_n\} = \{(k, \pi_i(k)) : k, i \in \mathbb{Z}_n\}. \tag{3.2}$$

Now we define

$$H_x = \{x_0 + x_s : s \in \mathbb{Z}_n\},\ H'_x = \{x_k + x_s : k, s \in \mathbb{Z}_n\},$$

$$G_x = \{x_n + x_{n+s} : s \in \mathbb{Z}_n\},\ G'_x = \{x_{n+k} + x_{n+s} : k, s \in \mathbb{Z}_n\}.$$

Therefore, $H_x \subseteq H'_x$ and $G_x \subseteq G'_x$.

As $G'_x = \{x_{n+k} + x_{n+\pi_i(k)} : k, i \in \mathbb{Z}_n\}$ from Equation 3.2 and $x_{n+k} + x_{n+\pi_i(k)} = x_0 + x_i$ from Equation 3.1, we have $G'_x = \{x_0 + x_i : i \in \mathbb{Z}_n\} = H_x$. Since A^T_X (the transpose of A_X) is also a compact Cauchy matrix generated by the vector $X' = (x_n, x_{n+1}, \ldots, x_{2n-1}, x_0, \ldots, x_{n-1})$, therefore for the same reason $H'_x = G_x$. Thus, we have $G'_x = H_x \subseteq H'_x = G_x \subseteq G'_x$.

So $H_x = H'_x$, which implies that H_x is closed under addition. Since H_x is finite, H_x is a subgroup of $\mathbb{F}_{2^r}$. Let $H = H_x$, by the definition of H_x and G_x, we arrive that $\{x_0, x_1, \ldots, x_{n-1}\} = x_0 + H$ and $\{x_n, x_{n+1}, \ldots, x_{2n-1}\} = x_n + G_x = x_n + H_x = x_n + H$.

Since A_X is a Cauchy matrix, we have $x_0 \notin x_n + H$. Therefore, $x_0 + x_n \notin H$. Here x_0 and x_n are playing the role of a and b respectively

For the converse part, we can proceed as Lemma 3.3. □

Lemma 3.3 presents a construction method for generating MDS matrices. It is important to note that these matrices may not be involutory. In the context of SPN, the decryption process typically requires the inverse of matrix A. If an efficient MDS matrix A used in encryption, happens to be involutory, then its inverse A^{-1} applied for decryption will also be efficient. In order to construct an involutory MDS matrix, we will consider the following lemma, which is also presented in [YMT97], but in a slightly different setting.

Lemma 3.5. *[GR13a, Lemma 8] Let $A = (a_{i,j})$ be an $n \times n$ matrix formed by Lemma*

3.3. Then, $A^2 = c^2 I$ where $c = \sum_{j=0}^{n-1} \frac{1}{l+x_j}$.

Proof. Let $A^2 = B = (b_{i,j})$. Since A is a symmetric matrix, we have $b_{i,j} = A_{row(i)} \cdot A_{row(j)}$. Therefore,

$$b_{i,i} = \sum_{k=0}^{n-1} \frac{1}{(l + x_i + x_k)^2} = \sum_{j=0}^{n-1} \frac{1}{(l + x_j)^2} = c^2$$

Similarly, for $i \neq j$,

$$b_{i,j} = \sum_{k=0}^{n-1} \frac{1}{(l + x_i + x_k)(l + x_j + x_k)} = \frac{1}{x_i + x_j} \sum_{k=0}^{n-1} \left(\frac{1}{l + x_i + x_k} + \frac{1}{l + x_j + x_k}\right) = 0.$$

Thus, $A^2 = c^2 I$. □

Corollary 3.3. *[GR13a, Corollary 5] If an $n \times n$ MDS matrix A is constructed from Lemma 3.3, $c^{-1}A$ is an involutory MDS matrix, where c is the sum of all elements of any row.*

Therefore, if the sum of the elements of any row of the matrix A of Lemma 3.3 is 1, A will be involutory.

Remark 3.5. *In [CJK15], authors proposed construction of involutory compact Cauchy matrix which directly follows from Corollary 3.3.*

The following is a example of a compact Cauchy matrix constructed using Lemma 3.3.

Example 3.3. *Let α be the primitive element of $\mathbb{F}_{2^4}$ whose constructing polynomial is $x^4 + x + 1$. Let $G = \{0, \alpha, \alpha^3, \alpha + \alpha^3\}$ and $l = \alpha^2$. Therefore,*

$$l + G = \left\{\alpha^2, \alpha + \alpha^2, \alpha^3 + \alpha^2, \alpha + \alpha^3 + \alpha^2\right\}.$$

Then the matrix

$$A = \begin{bmatrix} \frac{1}{(\alpha^2)} & \frac{1}{(\alpha+\alpha^2)} & \frac{1}{(\alpha^3+\alpha^2)} & \frac{1}{(\alpha+\alpha^3+\alpha^2)} \\ \frac{1}{(\alpha+\alpha^2)} & \frac{1}{(\alpha^2)} & \frac{1}{(\alpha+\alpha^3+\alpha^2)} & \frac{1}{(\alpha^3+\alpha^2)} \\ \frac{1}{(\alpha^3+\alpha^2)} & \frac{1}{(\alpha+\alpha^3+\alpha^2)} & \frac{1}{(\alpha^2)} & \frac{1}{(\alpha+\alpha^2)} \\ \frac{1}{(\alpha+\alpha^3+\alpha^2)} & \frac{1}{(\alpha^3+\alpha^2)} & \frac{1}{(\alpha+\alpha^2)} & \frac{1}{(\alpha^2)} \end{bmatrix}$$

$$= \begin{bmatrix} \alpha^3+\alpha^2+1 & \alpha^2+\alpha+1 & \alpha^3+\alpha & \alpha+1 \\ \alpha^2+\alpha+1 & \alpha^3+\alpha^2+1 & \alpha+1 & \alpha^3+\alpha \\ \alpha^3+\alpha & \alpha+1 & \alpha^3+\alpha^2+1 & \alpha^2+\alpha+1 \\ \alpha+1 & \alpha^3+\alpha & \alpha^2+\alpha+1 & \alpha^3+\alpha^2+1 \end{bmatrix}$$

is MDS matrix with exactly 4 distinct elements but not involutory. Sum of any row is $\alpha^3+\alpha^2+1+\alpha^2+\alpha+1+\alpha^3+\alpha+\alpha+1 = \alpha+1$ *and* $\frac{1}{(\alpha+1)^2}A^2 = I$*. Hence,* $\frac{1}{(\alpha+1)}A$ *is an involutory MDS matrix.*

Remark 3.6. *To ensure implementation-friendly designs in block ciphers and hash functions, it is desirable to maximize the number of ones in MDS matrices. In the construction of an* $n \times n$ *matrix* A *based on Lemma 3.3, each element occurs exactly* n *times (as stated in Lemma 3.4). Thus, the maximum number of ones that can appear in* A *is* n*. It should be noted that by multiplying* A *by the inverse of one of its entries, we can convert it to have the maximum number of ones without affecting the MDS property. However, this technique only guarantees ones in every row and does not provide control over the other* $n-1$ *elements. Also, if* A *is an involutory MDS matrix, this conversion will disrupt its involutory property.*

Remark 3.7. *In the paper [JV05b], the authors proposed a method to construct efficient MDS matrices by maximizing the number of occurrences of the element* 1 *and minimizing the number of occurrences of other distinct elements from the set* $\mathbb{F}_{2^r}^*$*. It is worth noting that by multiplying each row of an* $n \times n$ *MDS matrix* A *by the inverse of the first element of the respective row, we obtain a new MDS matrix* A' *with all* 1*'s in the first column (refer to Lemma 2.7). Similarly, by multiplying each column of the resulting* A' *matrix (starting from the second column) by the inverse of the first element of the respective column, we obtain a new MDS matrix* A'' *with all* 1*'s in the first row and first column. Consequently, the number of* 1*'s in this matrix* A'' *is* $2n-1$*. However, it should be noted that although* A'' *contains the maximum possible number of* 1*'s achievable starting from the MDS matrix* A*, the number of other distinct terms in* A'' *may exceed* $n-1$*. Moreover, in the case where the order of the matrix is even, the resulting matrix* A'' *will never be involutory.*

Now, we will discuss how to construct a Hadamard MDS matrix using Cauchy matrices. To do this, we require the following results.

Fact 3.1. *A* $2^n \times 2^n$ *matrix* $H = (h_{i,j})$ *is Hadamard in* $\mathbb{F}_{2^r}$ *if and only if* $h_{i,j} = h_{i+k,j+k}$ *and* $h_{i,j+k} = h_{j+k,i}$ *for* $0 \leq i, j \leq 2^{m-1}-1$ *and* $k = 2^{m-1}$ *where* $1 \leq m \leq n$*.*

Lemma 3.6. *Let $H = (h_{i,j})$ be a $2^n \times 2^n$ matrix whose first row is $(h_0, h_1, \ldots, h_{2^n-1})$, then H is Hadamard if and only if $h_{i,j} = h_{i\oplus j}$, where in $i \oplus j$, i and j are the n-bit binary representation of i and j respectively.*

Proof. If part: Suppose that $h_{i,j} = h_{i\oplus j}$. Then for $1 \leq m \leq n$, $0 \leq i, j \leq 2^{m-1} - 1$ and $k = 2^{m-1}$, we have $(i+k) \oplus (j+k) = i \oplus j$. Therefore, $h_{i,j} = h_{i\oplus j} = h_{(i+k)\oplus(j+k)} = h_{i+k,j+k}$. Again $h_{i,(j+k)} = h_{i\oplus(j+k)} = h_{(j+k)\oplus i} = h_{(j+k),i}$. Therefore, by Fact 3.1, H is a Hadamard matrix.

Only if part: Suppose that H is a Hadamard matrix of order 2^n. We have to show that $h_{i,j} = h_{i\oplus j}$ for $0 \leq i, j \leq 2^n - 1$. We will prove this by using the principle of mathematical induction. For $n = 1$,

$$H = \begin{bmatrix} h_0 & h_1 \\ h_1 & h_0 \end{bmatrix}.$$

Here $h_{0,0} = h_0 = h_{0\oplus 0}$, $h_{1,1} = h_0 = h_{1\oplus 1}$, $h_{0,1} = h_1 = h_{0\oplus 1}$ and $h_{1,0} = h_1 = h_{1\oplus 0}$. Therefore, the result is true for $n = 1$. Suppose that the result is true for $n = l$. Now suppose that H is a Hadamard matrix of order 2^{l+1} with the first row $(h_0, h_1, \ldots, h_{2^{l+1}-1})$. Since H is Hadamard, $H = \begin{bmatrix} U & V \\ V & U \end{bmatrix}$, where $U = (u_{i,j})$ and $V = (v_{i,j})$ are the Hadamard matrices of order 2^l with the first row $(h_0, h_1, \ldots, h_{2^l-1})$ and $(h_{2^l}, h_{2^l+1}, \ldots, h_{2^{l+1}-1})$ respectively. Now for $0 \leq i, j \leq 2^l - 1$ and $k = 2^l$, $h_{i,j} = u_{i,j}$ and $h_{i,j+k} = v_{i,j}$. Now by induction hypothesis as U is Hadamard, $h_{i,j} = h_{i\oplus j}$ for $0 \leq i, j \leq 2^l - 1$. Since $k = 2^l$ and $0 \leq i, j \leq 2^l - 1$, we have $i \oplus j = (i + k) \oplus (j + k)$. Therefore, from Fact 3.1, we have $h_{i+k,j+k} = h_{i,j} = h_{i\oplus j} = h_{(i+k)\oplus(j+k)}$. Similarly, for V, it can be checked by applying induction hypothesis that $h_{i,j+k} = h_{i\oplus(j+k)}$. From Fact 3.1 we have, $h_{(j+k),i} = h_{i,(j+k)} = h_{i\oplus(j+k)} = h_{(j+k)\oplus i}$. Therefore, $h_{i,j} = h_{i\oplus j}$ for $0 \leq i, j \leq 2^{l+1} - 1$. Therefore, by induction the result is true for all n. □

Note that a Hadamard matrix can be represented by its first row. We will denote the Hadamard matrix with its first row $(h_0, h_1, \ldots, h_{2^n-1})$ as $had(h_0, h_1, \ldots, h_{2^n-1})$. Because of the structure of Hadamard matrices the following fact is easy to verify.

Fact 3.2. *Let $H = (h_{i,j})$ be a square matrix of order 2^n and f be a bijection such that $f(h_{i,j}) = h'_{i,j}$. Then H is Hadamard if and only if $H' = (h'_{i,j})$ is Hadamard.*

Lemma 3.7. *Let $G = \{x_0, x_1, \ldots, x_{2^n-1}\}$ be an additive subgroup of $\mathbb{F}_{2^r}$ which is a linear span of n linearly independent elements $\{x_1, x_2, x_{2^2}, \ldots, x_{2^{n-1}}\}$ such that $x_i = \sum_{k=0}^{n-1} i_k x_{2^k}$ where $(i_{n-1}, \ldots, i_1, i_0)$ is the binary representation of i. Then $x_i + x_j = x_{i\oplus j}$.*

Proof. Suppose $(i_{n-1}, \ldots, i_1, i_0)$ and $(j_{n-1}, \ldots, j_1, j_0)$ are the binary representation of i and j respectively. Therefore, $x_i = i_0x_1 + i_1x_2 + i_2x_{2^2} + \ldots + i_{n-1}x_{2^{n-1}}$ and $x_j = j_0x_1 + j_1x_2 + j_2x_{2^2} + \ldots + j_{n-1}x_{2^{n-1}}$. Therefore, $x_i + x_j = (i_0 + j_0)x_1 + (i_1 + j_1)x_2 + (i_2 + j_2)x_{2^2} + \ldots + (i_{n-1} + j_{n-1})x_{2^{n-1}} = x_{i\oplus j}$. □

Remark 3.8. *The additive subgroup $G = \{x_0, \ldots, x_{2^n-1}\}$ in Lemma 3.7 is constructed by the linear combination of n linearly independent elements labeled x_1, x_2, x_{2^2}, …, $x_{2^{n-1}}$. Once x_1, x_2, x_{2^2}, …, $x_{2^{n-1}}$ have been fixed every other element $x_i \in G$ will be fixed to satisfy $x_i + x_j = x_{i\oplus j}$.*

Now we are ready to provide the following corollary from [GR13a].

Corollary 3.4. *[GR13a, Fact 9] Let $G = \{x_0, x_1, \ldots, x_{2^n-1}\}$ be an additive subgroup of $\mathbb{F}_{2^r}$ with $x_i + x_j = x_{i\oplus j}$, where in $i \oplus j$, i and j are the n-bit binary representation of i and j respectively. Then for $l \in \mathbb{F}_{2^r} \setminus G$, the matrix $H' = (h'_{i,j}) = (\frac{1}{l+x_{i\oplus j}})$ is Hadamard.*

Proof. Consider the matrix $H = (h_{i,j})$ of order 2^n, where $h_{i,j} = x_{i\oplus j}$ for $0 \leq i, j \leq 2^n - 1$. Therefore, the first row of H is $(x_0, x_1, \ldots, x_{2^n-1})$. Then from Lemma 3.6, $H = had(x_0, x_1, x_2, \ldots, x_{2^n-1})$. Since $0 \notin l + G$, we have $l + x_{i\oplus j} = l + x_i + x_j \neq 0$. Now consider the bijection $f(h_{i,j}) = \frac{1}{l+h_{i,j}}$. Therefore, by Fact 3.2, H' is a Hadamard matrix. □

Lemma 3.8. *[GR13a, Theorem 4] Let $G = \{x_0, x_1, \ldots, x_{2^n-1}\}$ be an additive subgroup of $\mathbb{F}_{2^r}$ which is a linear span of n linearly independent elements $\{x_1, x_2, x_{2^2}, \ldots, x_{2^{n-1}}\}$ such that $x_i = \sum_{k=0}^{n-1} i_k x_{2^k}$ where $(i_{n-1}, \ldots, i_1, i_0)$ is the binary representation of i. Let $y_i = l + x_i$ for $0 \leq i \leq 2^n - 1$ where $l \in \mathbb{F}_{2^r} \setminus G$. Then the matrix $A = (a_{i,j})$, where $a_{i,j} = \frac{1}{(x_i+y_j)}$ is a Hadamard MDS matrix.*

Proof. Consider the matrix $H = (h_{i,j}) = (x_i + x_j)$. Then $h_{i,j} = x_{i\oplus j}$. Therefore, by Lemma 3.6, H is Hadamard. Now $a_{i,j} = \frac{1}{(x_i+y_j)} = \frac{1}{(l+x_i+x_j)} = \frac{1}{l+x_{i\oplus j}}$. Thus, from Corollary 3.4, A is Hadamard. Again by Lemma 3.3, A is MDS. Hence, A is a Hadamard MDS matrix. □

Remark 3.9. *We will call this construction as Cauchy based construction of type 4. Also note that the matrix constructed using Lemma 3.8 may not be an involutory. Whereas $\frac{1}{c}A$ is a Hadamard involutory MDS matrix, where c is the sum of the elements of any row. ANUBIS [BR00a] uses Hadamard involutory matrix which was constructed by exhaustive search and not by Lemma 3.8.*

Example 3.4. *Let α be the primitive element of $\mathbb{F}_{2^4}$ whose constructing polynomial is x^4+x+1. Let $G=\{x_0=0,\ x_1=\alpha,\ x_2=\alpha^3,\ x_3=\alpha+\alpha^3\}$ be the additive group spanned by $\{x_1=\alpha,\ x_2=\alpha^3\}$ and let $l=\alpha^2$. Therefore,*

$$y_0=\alpha^2,\ y_1=\alpha+\alpha^2,\ y_2=\alpha^3+\alpha^2 \text{ and } y_3=\alpha+\alpha^3+\alpha^2.$$

Then the matrix

$$A=\begin{bmatrix} \frac{1}{(\alpha^2)} & \frac{1}{(\alpha+\alpha^2)} & \frac{1}{(\alpha^3+\alpha^2)} & \frac{1}{(\alpha+\alpha^3+\alpha^2)} \\ \frac{1}{(\alpha+\alpha^2)} & \frac{1}{(\alpha^2)} & \frac{1}{(\alpha+\alpha^3+\alpha^2)} & \frac{1}{(\alpha^3+\alpha^2)} \\ \frac{1}{(\alpha^3+\alpha^2)} & \frac{1}{(\alpha+\alpha^3+\alpha^2)} & \frac{1}{(\alpha^2)} & \frac{1}{(\alpha+\alpha^2)} \\ \frac{1}{(\alpha+\alpha^3+\alpha^2)} & \frac{1}{(\alpha^3+\alpha^2)} & \frac{1}{(\alpha+\alpha^2)} & \frac{1}{(\alpha^2)} \end{bmatrix}$$

$$=\begin{bmatrix} \alpha^3+\alpha^2+1 & \alpha^2+\alpha+1 & \alpha^3+\alpha & \alpha+1 \\ \alpha^2+\alpha+1 & \alpha^3+\alpha^2+1 & \alpha+1 & \alpha^3+\alpha \\ \alpha^3+\alpha & \alpha+1 & \alpha^3+\alpha^2+1 & \alpha^2+\alpha+1 \\ \alpha+1 & \alpha^3+\alpha & \alpha^2+\alpha+1 & \alpha^3+\alpha^2+1 \end{bmatrix}$$

is a Hadamard MDS matrix but not involutory. Sum of the elements of any row is $\alpha+1$ and hence $\frac{1}{\alpha+1}A$ is involutory.

Remark 3.10. *So far we have type 1, type 2, type 3 and type 4 Cauchy based constructions by Lemma 3.1, Remark 3.1, Lemma 3.3 and Lemma 3.8 respectively. Similarly, in the next section, we will discuss type 1, type 2, type 3 and type 4 Vandermonde based constructions to construct MDS matrices.*

Remark 3.11. *If $A=(a_{i,j})$ is a Cauchy matrix, where $a_{i,j}=\frac{1}{x_i+y_j}$ and $x_i+y_j\neq 0$ for $0\leq i,j\leq n-1$ then for any two nonsingular diagonal matrices $D_1=diag(c_0,c_1,\ldots,c_{n-1})$ and $D_2=diag(d_0,d_1,\ldots,d_{n-1})$, the matrix $D_1AD_2=(\frac{c_id_j}{x_i+y_j})$ is called generalized Cauchy matrix. We know from Corollary 2.4 that if A is MDS then D_1AD_2 is MDS. Also note that even if a Cauchy matrix is not involutory, its corresponding generalized Cauchy matrix can be made involutory for a suitable choice of D_1 and D_2. We will discuss it later in Section 3.4.*

We provide another construction of an MDS matrix, which is slightly modified version of [MRS12, RS85] and is closely related to Cauchy based construction.

Theorem 3.2. *[RS85, Theorem 3] Suppose $q=2^r$ and γ is an arbitrary primitive element of the field $\mathbb{F}_q$. Let S_q be a triangular array whose coefficients are constants along skew diagonal in a Hankel matrix fashion defined as*

$$S_q = \begin{array}{cccccc} a_1 & a_2 & a_3 & \dots & a_{q-3} & a_{q-2} \\ a_2 & a_3 & a_4 & \dots & a_{q-2} & \\ a_3 & a_4 & \dots & a_{q-2} & & \\ a_4 & \dots & & & & \\ \vdots & \vdots & & & & \\ a_{q-3} & a_{q-2} & & & & \\ a_{q-2} & & & & & \end{array}$$

where $a_i = (1-\gamma^i)^{-1}$ *for* $1 \le i \le q-2$. *Then every square submatrix of* S_q *is nonsingular and hence MDS.*

Proof. For $1 \le i \le q-2$ and $1 \le j \le q-i-1$, let $s_{i,j}$ be the entries of S_q. Thus,

$$\begin{aligned} s_{i,j} = a_{i+j-1} &= \frac{1}{1-\gamma^{i+j-1}}, \text{ for } 1 \le i \le q-2 \text{ and } 1 \le j \le q-i-1 \\ &= \frac{1}{1-\frac{\gamma^j}{\gamma^{-(i-1)}}}. \end{aligned}$$

Consider the vector $x = (x_1, x_2, ..., x_{q-2})$ and $y = (y_1, y_2, ..., y_{q-2})$ defined by

$$x_i = -\gamma^{-(i-1)}, \; y_j = \gamma^j, \text{ for } 1 \le i \le q-2 \text{ and } 1 \le j \le q-2.$$

It is easy to check that x_i's and y_j's are distinct and $x_i + y_j \neq 0$ for $i + j \le q-1$. It can be readily verified that

$$s_{i,j} = \frac{x_i}{x_i + y_j}, \text{ for } 1 \le i \le q-2 \text{ and } \; 1 \le j \le q-i-1.$$

Since all the x_i's are distinct and nonzero, all the y_j's are distinct and $x_i + y_j \neq 0$ for i and j in the defined ranges, we conclude that every square submatrix of S_q is a nonsingular generalized Cauchy matrix. □

We close this section by providing an interconnection between Reed-Solomon code and generalized Cauchy matrix [RS85, Theorem 1].

Theorem 3.3. *[RS85, RL89] A matrix of the form* $G = [I \mid A]$ *over a finite field* $\mathbb{F}_q$ *generates a generalized Reed-Solomon code if and only if* $A = (a_{i,j})$ *is a generalized Cauchy matrix i.e.* $a_{i,j} = \frac{c_i d_j}{x_i + y_j}$ *for* $0 \le i, j \le n-1$, *where the* x_i, y_j*'s are* $2n$ *distinct elements of* $\mathbb{F}_q$, *such that* $x_i + y_j \neq 0$ *for all* i *and* j *and* $c_i, d_j \neq 0$.

3.3 Constructing MDS Matrices from Vandermonde Matrices

Application of Vandermonde matrices for constructing MDS codes is widely available in the literature [GR13a, LF04a, LF04b, MRS12, SDMO12]. Vandermonde matrices over a finite field can have singular square submatrices (see Fact 2.10). Consequently, these matrices by themselves need not be MDS over a finite field. Lacan and Fimes [LF04a, LF04b] used two Vandermonde matrices to build an MDS matrix. Later, Sajadieh et al. [SDMO12] used similar method to find an MDS matrix that is also involutory. We will mainly discuss [GR13a, LF04b, PSA+18, SDMO12] in this section.

Theorem 3.4. *[LF04b, Theorem 2] Let $V_1 = vand(a_0, a_1, \ldots, a_{n-1})$ and $V_2 = vand(b_0, b_1, \ldots, b_{n-1})$ be two Vandermonde matrices such that a_i, b_j are $2n$ distinct elements from some field. Then the matrices $V_1^{-1}V_2$ and $V_2^{-1}V_1$ are such that any square submatrix of them is nonsingular and hence MDS matrices.*

Proof. Let us denote by U the $n \times 2n$ matrix $[V_1 \mid V_2]$. Consider the product $W = V_1^{-1}U = [I \mid A]$ where $A = V_1^{-1}V_2$. Now, we prove that A does not contain any singular submatrix.

Every $n \times n$ submatrix of U is nonsingular because it is also a Vandermonde matrix built from n distinct elements. Then any $n \times n$ submatrix of W is also nonsingular for it is the product of V_1^{-1} and the corresponding nonsingular $n \times n$ submatrix of U. Now from Remark 3.12 (written below), the code defined by the generator matrix $[I \mid A]$ is an MDS code. Thus, $V_1^{-1}V_2$ is an MDS matrix. For $V_2^{-1}V_1$ the proof is identical. □

Remark 3.12. *In the above theorem we have used Corollary 3 of [MS77, Page 319]: A generator matrix $G = [I \mid A]$ generates an $[2n, n, n+1]$ MDS code if and only if every set of n columns of G is linearly independent.*

Remark 3.13. *We will call the construction using Theorem 3.4 as Vandermonde based construction of type 1. Note that in Cauchy based construction of type 1 (see Lemma 3.1), a extra condition $x_i + y_j \neq 0$ for $0 \leq i, j \leq n-1$ is needed.*

Remark 3.14. *Some authors [GR13a, SDMO12] use notation $vand(a_0, a_1, a_2, \ldots, a_{n-1}) = A^T$, where A is as defined in Definition 2.23. With this notation $V_1V_2^{-1}$ and $V_2V_1^{-1}$ will be MDS.*

Example 3.5. *Let α be the primitive element of $\mathbb{F}_{2^4}$ whose constructing polynomial is $x^4 + x + 1$. Consider the Vandermonde matrices $V_1 = vand(0, \alpha^4, \alpha^8)$ and $V_2 = vand(1, \alpha^3, \alpha^5)$. Then the matrix*

$$V_1^{-1}V_2 = \begin{bmatrix} \alpha^3 + \alpha^2 & \alpha^2 + 1 & 1 \\ \alpha^2 + 1 & \alpha^3 + \alpha + 1 & 1 \\ \alpha^3 & \alpha^3 + \alpha^2 + \alpha + 1 & 1 \end{bmatrix}$$

is MDS but not involutory.

In [SDMO12], the authors showed that for two Vandermonde matrices $V_1 = vand(a_0, a_1, \ldots, a_{n-1})$ and $V_2 = vand(b_0, b_1, \ldots, b_{n-1}) = vand(l + a_0, l + a_1, \ldots, l + a_{n-1})$, where l is an arbitrary nonzero element in $\mathbb{F}_{2^r}$, the matrix $V_1^{-1}V_2$ is involutory (see also Remark 3.15). Again if a_i's and b_i's are $2n$ different values, then by Theorem 3.4, $V_1^{-1}V_2$ will be involutory MDS matrix. Corollary 3.5 states this result formally which is a direct consequence of Theorem 3.4 and Theorem 3.6. Theorem 3.5 is an intermediate result for proving Theorem 3.6.

Remark 3.15. *$V_1^{-1}V_2$ is involutory if and only if $V_1^{-1}V_2 = V_2^{-1}V_1$*

Theorem 3.5. *[SDMO12, Theorem 3] If $V_1 = vand(a_0, a_1, \ldots, a_{n-1})$ and $V_2 = vand(b_0, b_1, \ldots, b_{n-1})$ are two invertible Vandermonde matrices such that $b_i = l + a_i$, then $V_2{V_1}^{-1}$ is lower triangular matrix whose nonzero elements are determined by powers of l.*

Proof. Let $V_1^{-1} = (t_{i,j})$ and $V = (v_{i,j}) = V_2 \cdot V_1^{-1}$, $0 \le i, j \le n-1$.
As $V_1 \cdot V_1^{-1} = I$, we have

$$V_{1_{row(0)}}.V^{-1}_{1_{column(0)}} = \sum_{i=0}^{n-1} t_{i,0} = 1 \text{ and}$$

$$V_{1_{row(k)}}.V^{-1}_{1_{column(0)}} = \sum_{i=0}^{n-1} a_i^k \cdot t_{i,0} = 0 \text{ for } 1 \le k \le n-1.$$

It can be checked that

$$v_{0,0} = V_{2_{row(0)}}.V^{-1}_{1_{column(0)}} = \sum_{i=0}^{n-1} t_{i,0} = 1,$$

$$v_{k,0} = V_{2_{row(k)}}.V^{-1}_{1_{column(0)}} = \sum_{i=0}^{n-1} b_i^k \cdot t_{i,0} = \sum_{i=0}^{n-1} (l + a_i)^k \cdot t_{i,0}$$
$$= \sum_{i=0}^{n-1} ({}^kC_0 a_i^k + {}^kC_1 a_i^{k-1} \cdot l + \ldots + {}^kC_{k-1} a_i \cdot l^{k-1} + {}^kC_k l^k) \cdot t_{i,0}$$
$$= \sum_{i=0}^{n-1} l^k \cdot t_{i,0} = l^k \text{ for } 1 \leq k \leq n-1.$$

So we have computed the 0-th column of V.

Again as $V_1 \cdot V_1^{-1} = I$,

$$V_{1_{row(0)}}.V^{-1}_{1_{column(1)}} = \sum_{i=0}^{n-1} t_{i,1} = 0,$$
$$V_{1_{row(1)}}.V^{-1}_{1_{column(1)}} = \sum_{i=0}^{n-1} a_i \cdot t_{i,1} = 1 \text{ and}$$
$$V_{1_{row(k)}}.V^{-1}_{1_{column(1)}} = \sum_{i=0}^{n-1} a_i^k \cdot t_{i,1} = 0 \text{ for } 2 \leq k \leq n-1.$$

Again it can be checked that

$$v_{0,1} = V_{2_{row(0)}}.V^{-1}_{1_{column(1)}} = \sum_{i=0}^{n-1} t_{i,1} = 0,$$
$$v_{1,1} = V_{2_{row(1)}}.V^{-1}_{1_{column(1)}} = \sum_{i=0}^{n-1} b_i \cdot t_{i,1} = \sum_{i=0}^{n-1} (l + a_i) \cdot t_{i,1} = \sum_{i=0}^{n-1} a_i \cdot t_{i,1} = 1 \text{ and}$$
$$v_{k,1} = V_{1_{row(k)}}.V^{-1}_{1_{column(1)}} = \sum_{i=0}^{n-1} b_i^k \cdot t_{i,1} = \sum_{i=0}^{n-1} (l + a_i)^k \cdot t_{i,1}$$
$$= \sum_{i=0}^{n-1} ({}^kC_0 a_i^k + {}^kC_1 a_i^{k-1} \cdot l + \ldots + {}^kC_{k-1} a_i \cdot l^{k-1} + {}^kC_k l^k) \cdot t_{i,1}$$
$$= \sum_{i=0}^{n-1} {}^kC_{k-1} a_i \cdot l^{k-1} \cdot t_{i,1}$$
$$= {}^kC_{k-1} \cdot l^{k-1} = {}^kC_1 \cdot l^{k-1} \text{ for } 2 \leq k \leq n-1.$$

So we have computed the 1-st column of V. Similarly,

$$v_{0,2} = v_{1,2} = 0,\ v_{2,2} = 1 \text{ and } v_{k,2} = {}^kC_2 \cdot l^{k-2} \text{ for } 3 \leq k \leq n-1,$$

$$v_{0,3} = v_{1,3} = v_{2,3} = 0,\ v_{3,3} = 1 \text{ and } v_{k,3} = {}^{k}C_3 \cdot l^{k-3} \text{ for } 4 \leq k \leq n-1,$$

and so on. Therefore, $V = V_2 \cdot V_1^{-1}$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 & \ldots\ldots & 0 & 0 \\ l & 1 & 0 & 0 & \ldots\ldots & 0 & 0 \\ l^2 & {}^{2}C_1 \cdot l & 1 & 0 & \ldots\ldots & 0 & 0 \\ l^3 & {}^{3}C_1 \cdot l^2 & {}^{3}C_2 \cdot l & 1 & \ldots\ldots & 0 & 0 \\ l^4 & {}^{4}C_1 \cdot l^3 & {}^{4}C_2 \cdot l^2 & {}^{4}C_3 \cdot l & \ldots\ldots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ldots\ldots & \vdots & \vdots \\ l^{n-1} & {}^{n-1}C_1 \cdot l^{n-2} & {}^{n-1}C_2 \cdot l^{n-3} & {}^{n-1}C_3 \cdot l^{n-4} & \ldots\ldots & l & 1 \end{bmatrix}.$$

Thus, $V_2V_1^{-1}$ is a lower triangular matrix. □

Theorem 3.6. *[SDMO12, Theorem 4] If $V_1 = vand(a_0, a_1, \ldots, a_{n-1})$ and $V_2 = vand(b_0, b_1, \ldots, b_{n-1})$ are two invertible Vandermonde matrices such that $a_i = l + b_i$, then $V_2V_1^{-1}V_2 = V_1$.*

Proof. Let $V = (v_{i,j}) = V_2 \cdot V_1^{-1}$. Note that $(V_1)_{i,j} = a_j^i$ and $(V_2)_{i,j} = b_j^i$. Therefore,

$$\begin{aligned} (V \cdot V_2)_{i,j} &= V_{row(i)} \cdot V_{2_{column(j)}} \\ &= l^i + {}^{i}C_1 l^{i-1} \cdot b_j + {}^{i}C_2 l^{i-2} \cdot b_j^2 + \ldots + {}^{i}C_{i-1} b_j^{i-1} + b_j^i \\ &= (l + b_j)^i = a_j^i. \end{aligned}$$

Therefore, $V_2V_1^{-1}V_2 = V_1$. □

Remark 3.16. *$V_2V_1^{-1}V_2 = V_1$ implies that $(V_1^{-1}V_2)^2 = I$ i.e. $V_1^{-1}V_2$ is involutory.*

Corollary 3.5. *[SDMO12, Corollary 1] If $V_1 = vand(a_0, a_1, \ldots, a_{n-1})$ and $V_2 = vand(b_0, b_1, \ldots, b_{n-1})$ are two invertible Vandermonde matrices in the field $\mathbb{F}_{2^r}$ satisfying the two properties $a_i = l + b_i$ and $a_i \neq b_j$, for $0 \leq i, j \leq n-1$, then $V_1^{-1}V_2$ is involutory MDS matrix.*

We will call this construction as Vandermonde based construction of type 2. This construction gives involutory MDS matrices. Whereas in Cauchy based construction of type 2 (see Remark 3.1) the constructed MDS matrix need not be involutory.

Example 3.6. *Let α be the primitive element of $\mathbb{F}_{2^4}$ whose constructing polynomial is $x^4 + x + 1$. Let $l = 1$, $x_0 = \alpha$, $x_1 = \alpha^2$, $x_2 = \alpha^3$ and $y_0 = 1 + \alpha$, $y_1 =$*

$1+\alpha^2$, $y_2 = 1+\alpha^3$. *Consider the Vandermonde matrices* $V_1 = vand(\alpha, \alpha^2, \alpha^3)$ *and* $V_2 = vand(1+\alpha, 1+\alpha^2, 1+\alpha^3)$. *Then the matrix*

$$V_1^{-1}V_2 = \begin{bmatrix} \alpha^3 & \alpha^3+1 & \alpha^3+1 \\ \alpha^3+\alpha^2+\alpha & \alpha^3+\alpha^2+\alpha+1 & \alpha^3+\alpha^2+\alpha \\ \alpha^2+\alpha+1 & \alpha^2+\alpha+1 & \alpha^2+\alpha \end{bmatrix}$$

is involutory MDS and

$$V_2V_1^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}$$

is a lower triangular matrix.

Remark 3.17. *Let* $G = \{x_0, x_1, \ldots, x_{n-1}\}$ *be an additive subgroup of* $\mathbb{F}_{2^r}$. *Let us consider the coset* $l+G$, $l \notin G$ *having elements* $y_j = l + x_j$, $j = 0, \ldots, n-1$. *If* $V_1 = vand(x_0, x_1, \ldots, x_{n-1})$ *and* $V_2 = vand(l+x_0, l+x_1, \ldots, l+x_{n-1})$, *then* $V_1^{-1}V_2$ *is involutory MDS matrix by Corollary 3.5. We will call this construction as Vandermonde based construction of type* 3. *Note its similarity of* x_i*'s and* y_j*'s of Cauchy based construction of type* 3 *using Lemma 3.3.*

In [SDMO12], authors defined Special Vandermonde matrix, which was restated differently but equivalently in [GR13a] as follows.

Definition 3.1. *[GR13a] Let* G *be an additive subgroup* $\{x_0, x_1, \ldots, x_{2^n-1}\}$ *of* $\mathbb{F}_{2^r}$ *of order* 2^n *which is a linear span of* n *linearly independent elements* $\{x_1, x_2, x_{2^2}, \ldots, x_{2^{n-1}}\}$ *such that* $x_i = \sum_{i=0}^{n-1} b_i x_{2^i}$ *where* $(b_{n-1}, \ldots, b_1, b_0)$ *is the binary representation of* i. *A Vandermonde matrix* $vand(y_0, y_1, \ldots, y_{2^n-1})$ *is called Special Vandermonde matrix if* $y_i = l + x_i$.

In [SDMO12, Corollary 2], authors provided a construction of Hadamard involutory MDS matrices using Special Vandermonde matrices which was generalized in [GR13a, Lemma 5]. We restate it in the following lemma.

Lemma 3.9. *[GR13a, Lemma 5] Let* $V_1 = vand(x_0, x_1, \ldots, x_{2^n-1})$ *and* $V_2 = vand(y_0, y_1, \ldots, y_{2^n-1})$ *be Special Vandermonde matrices in* $\mathbb{F}_{2^r}$ *where* $y_i = x_0 + y_0 + x_i$ *and* $y_0 \notin \{x_0, x_1, \ldots, x_{2^n-1}\}$, *then* $V_1^{-1}V_2$ *is involutory Hadamard MDS matrix.*

The proof of Corollary 2 in [SDMO12] spanned multiple pages. The authors of [GR13a] proposed an alternative and much simpler proof of the above

lemma [GR13a, Corollary 8] and we will provide another proof in Section 3.4, Corollary 3.7.

We will call the construction of Lemma 3.9 as Vandermonde based construction of type 4. Note that the Cauchy based construction of type 4 using Lemma 3.8 also provides Hadamard MDS matrix but it may not be involutory.

Example 3.7. *Let α be the primitive element of $\mathbb{F}_{2^4}$ with $\alpha^4 + \alpha + 1 = 0$. Let $G = \{0, \alpha, \alpha^3, \alpha + \alpha^3\}$ and let $y_0 = \alpha^2$. Therefore, $x_0 + y_0 = \alpha^2$. Consider the matrices $V_1 = vand(0,\ \alpha,\ \alpha^3,\ \alpha + \alpha^3)$ and $V_2 = vand(\alpha^2,\ \alpha + \alpha^2,\ \alpha^3 + \alpha^2,\ \alpha + \alpha^3 + \alpha^2)$. The matrix*

$$V_1^{-1}V_2 = \begin{bmatrix} \alpha^3+\alpha & \alpha^3+\alpha^2 & \alpha^2+\alpha & 1 \\ \alpha^3+\alpha^2 & \alpha^3+\alpha & 1 & \alpha^2+\alpha \\ \alpha^2+\alpha & 1 & \alpha^3+\alpha & \alpha^3+\alpha^2 \\ 1 & \alpha^2+\alpha & \alpha^3+\alpha^2 & \alpha^3+\alpha \end{bmatrix}$$

is Hadamard involutory MDS matrix.

Remark 3.18. *Lemma 3.8 provides Hadamard MDS matrix, Lemma 3.9 provides Hadamard MDS matrix which is also involutory. For the sake of efficiency, the Hadamard involutory MDS matrix used in ANUBIS block cipher [BR00a] was constructed by search method. The authors of [SKOP15] discussed the constructions of Hadamard MDS matrices by search methods in details. To reduce the search space they defined an equivalence classes of Hadamard matrices in terms of branch number. In [LS16], authors discussed the similarity between equivalence classes of Hadamard matrices and equivalence classes of circulant matrices. We will discuss equivalence classes of circulant matrices in Section 3.5.*

Theorem 3.7. *Let M be an MDS matrix and D be a nonsingular diagonal matrix. Then, DMD^{-1} will also be an MDS matrix. If $M^2 = cI$ for some constant c, then $(DMD^{-1})^2 = cI$.*

Proof. From Corollary 2.4, DMD^{-1} is MDS. Let $B = DMD^{-1}$, then

$$B^2 = DMD^{-1}DMD^{-1} = DM^2D^{-1} = D(cI)D^{-1} = cI.$$

□

In [PSA+18], authors proposed a new form of matrix, which they called generalized Hadamard matrix (GHadamard matrix) and provided several efficient MDS matrices. If H is a Hadamard matrix then DHD^{-1} is called a GHadamard matrix, where D is a nonsingular matrix. The underline idea of their construction is provided in Theorem 3.7. Note, H is involutory if and only if DHD^{-1} is involutory.

Remark 3.19. *In this section we discussed type* 1*, type* 2*, type* 3 *and type* 4 *Vandermonde based construction by Theorem 3.4, Corollary 3.5, Remark 3.17 and Lemma 3.9. Recall Remark 3.10 for type* 1*, type* 2*, type* 3 *and type* 4 *Cauchy based construction.*

Let M and V represent the MDS matrices obtained from the Cauchy based construction and Vandermonde based construction, respectively. In the next section we will show that they are related by $D_1MD_2 = V$, where D_1 and D_2 are nonsingular diagonal matrices.

3.4 Interconnection between Vandermonde Based Construction and Cauchy Based Construction

Till now, we discussed four types of MDS matrix constructions using Cauchy and Vandermonde matrices. In this section we provide a nontrivial interconnection between Cauchy based constructions and Vandermonde based constructions. Let $x_0, x_1, x_2, \ldots, x_{n-1}$ and $y_0, y_1, y_2, \ldots, y_{n-1}$ be $2n$ distinct elements from $\mathbb{F}_{2^r}$ with $x_i + y_j \neq 0$ for all $0 \leq i, j \leq n-1$. Consider the matrices $V_1 = vand(x_0, x_1, x_2, \ldots, x_{n-1})$, $V_2 = vand(y_0, y_1, y_2, \ldots, y_{n-1})$ and $M = (m_{i,j})$, where $m_{i,j} = \frac{1}{x_i + y_j}$. Then we know that $V_1^{-1}V_2$, $V_2^{-1}V_1$ *and* M are MDS matrices.

Now we will prove that the type 1 Vandermonde based construction is equivalent to type 1 Cauchy based construction. Type 2, type 3 and type 4 are just the special cases. Note that Gupta et al. [GR13a, Theorem 5] proved the equivalence for type 4 construction which is a particular case of Theorem 3.8.

Theorem 3.8. *Suppose V_1, V_2 and M are as defined above and $V_1^{-1} = (b_{i,j})$, $0 \leq i, j \leq n-1$, then $D_1MD_2 = V_1^{-1}V_2$, where*

$$D_1 = diag(b_{0,n-1}, b_{1,n-1}, b_{2,n-1}, \ldots, b_{n-1,n-1}) \text{ and}$$

$$D_2 = diag(\prod_{k=0}^{n-1}(x_k + y_0), \prod_{k=0}^{n-1}(x_k + y_1), \prod_{k=0}^{n-1}(x_k + y_2), \ldots, \prod_{k=0}^{n-1}(x_k + y_{n-1})).$$

Proof. Consider the polynomial

$$P_i(x) = b_{i,0} + b_{i,1}x + b_{i,2}x^2 + \ldots + b_{i,n-1}x^{n-1} = \sum_{k=0}^{n-1} b_{i,k}x^k$$

whose coefficients are the elements of the i-th row of V_1^{-1}. Consider the (i,j)-th element of $V_1^{-1}V_2$,

$$(V_1^{-1}V_2)_{i,j} = \sum_{k=0}^{n-1} b_{i,k} \cdot (V_2)_{k,j} = \sum_{k=0}^{n-1} b_{i,k} \cdot y_j^k = P_i(y_j). \tag{3.3}$$

We will prove that $(D_1MD_2)_{i,j} = (V_1^{-1}V_2)_{i,j}$. Now

$$\begin{aligned}(D_1MD_2)_{i,j} &= (D_1)_{i,i} \cdot m_{i,j} \cdot (D_2)_{j,j} \\ &= b_{i,n-1} \cdot \frac{1}{x_i + y_j} \cdot (\prod_{k=0}^{n-1} (x_k + y_j)) \\ &= b_{i,n-1}(x_0 + y_j)(x_1 + y_j)\dots(x_{i-1} + y_j)(x_{i+1} + y_j)\dots(x_{n-1} + y_j).\end{aligned} \tag{3.4}$$

The i-th row of $V_1^{-1}V_1$

$$\begin{aligned}&= \begin{bmatrix} b_{i,0} & b_{i,1} & \dots & b_{i,n-1} \end{bmatrix} \cdot V_1 \\ &= \begin{bmatrix} \sum_{k=0}^{n-1} b_{i,k}x_0^k & \sum_{k=0}^{n-1} b_{i,k}x_1^k & \dots & \sum_{k=0}^{n-1} b_{i,k}x_{n-1}^k \end{bmatrix} \\ &= \begin{bmatrix} P_i(x_0) & P_i(x_1) & \dots & P_i(x_{n-1}) \end{bmatrix}.\end{aligned}$$

As $V_1^{-1}V_1 = I$, we have $P_i(x_i) = 1$ and $P_i(x_j) = 0$ for $i \neq j$ i.e. $x_0, x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_{n-1}$ are the roots of $P_i(x)$. Therefore,

$$P_i(x) = b_{i,n-1}(x + x_0)(x + x_1)\dots(x + x_{i-1})(x + x_{i+1})\dots(x + x_{n-1}). \tag{3.5}$$

From Equation 3.4 and Equation 3.5, we have $(D_1MD_2)_{i,j} = P_i(y_j)$ and from Equation 3.3 we have $(V_1^{-1}V_2)_{i,j} = P_i(y_j)$. Therefore, $(D_1MD_2)_{i,j} = (V_1^{-1}V_2)_{i,j}$. □

Remark 3.20. *In type 2 construction, $V_1^{-1}V_2$ is involutory but M is not involutory. But we can make it involutory by D_1MD_2, where D_1 and D_2 are the two nonsingular diagonal matrices as defined in Theorem 3.8. For example consider $x_0 = 0$, $x_1 = \alpha^4$, $x_2 = \alpha^8$ and $y_i = \alpha + x_i$, over $\mathbb{F}_{2^4}$ whose constructing polynomial is $x^4 + x + 1$ with α a primitive element. Then*

$$V_1^{-1}V_2 = \begin{bmatrix} \alpha^3 + \alpha^2 + 1 & \alpha^3 + \alpha^2 & \alpha^3 + \alpha^2 \\ \alpha^2 & \alpha^2 + 1 & \alpha^2 \\ \alpha^3 & \alpha^3 & \alpha^3 + 1 \end{bmatrix}$$

is an involutory MDS matrix. But the Cauchy matrix

$$M = \begin{bmatrix} \frac{1}{\alpha} & \frac{1}{\alpha+\alpha^4} & \frac{1}{\alpha+\alpha^8} \\ \frac{1}{\alpha+\alpha^4} & \frac{1}{\alpha} & \frac{1}{\alpha+\alpha^4+\alpha^8} \\ \frac{1}{\alpha+\alpha^8} & \frac{1}{\alpha+\alpha^4+\alpha^8} & \frac{1}{\alpha} \end{bmatrix} = \begin{bmatrix} \alpha^3+1 & 1 & \alpha^2+\alpha \\ 1 & \alpha^3+1 & \alpha^3+\alpha^2+1 \\ \alpha^2+\alpha & \alpha^3+\alpha^2+1 & \alpha^3+1 \end{bmatrix}$$

is not involutory. Now

$$V_1^{-1} = \begin{bmatrix} 1 & \alpha^2+1 & \alpha^3 \\ 0 & \alpha^3+1 & \alpha^3+\alpha^2 \\ 0 & \alpha^3+\alpha^2 & \alpha^2 \end{bmatrix}.$$

Therefore, $D_1 = diag(c_0, c_1, c_2)$ *and* $D_2 = diag(d_0, d_1, d_2)$ *where* $c_0 = \alpha^3$, $c_1 = \alpha^3+\alpha^2$, $c_2 = \alpha^2$, $d_0 = \alpha\cdot(\alpha+\alpha^4)\cdot(\alpha+\alpha^8) = \alpha^3+\alpha^2+\alpha$, $d_1 = (\alpha+\alpha^4)\cdot\alpha\cdot(\alpha+\alpha^4+\alpha^8) = \alpha^3$ *and* $d_2 = (\alpha+\alpha^8)\cdot(\alpha+\alpha^4+\alpha^8)\cdot\alpha = \alpha^3+\alpha^2+1$. *Now it is easy to check that* $D_1 M D_2 = V_1^{-1}V_2$. *Therefore, the generalized Cauchy matrix* $D_1 M D_2$ *is involutory.*

Cauchy based construction of type 3 provides compact MDS matrix (see Lemma 3.4). In Corollary 3.6 we will show that Vandermonde based construction of type 3 also provides compact MDS matrix. To prove this we need the following lemmas.

Let $\{x_0, x_1, \ldots, x_{n-1}\}$ be an additive subgroup G of $\mathbb{F}_{2^r}$ where $x_0 = 0$, $V_1 = vand(x_0, x_1, \ldots, x_{n-1})$ and

$$V_1^{-1} = \begin{bmatrix} b_{0,0} & b_{0,1} & \ldots & b_{0,n-1} \\ b_{1,0} & b_{1,1} & \ldots & b_{1,n-1} \\ \vdots & \vdots & \vdots & \vdots \\ b_{n-1,0} & b_{n-1,1} & \ldots & b_{n-1,n-1} \end{bmatrix}, \text{ where } b_{i,j} \in \mathbb{F}_{2^r}$$

and let γ be the product of all nonzero elements in G i.e. $\gamma = \prod_{i=1}^{n-1} x_i$.

Lemma 3.10. *Let* V_1 *and* γ *are as defined above, then* $\det(V_1) = \gamma^{\frac{n}{2}}$.

Proof. We have $\det(V_1) = \prod_{k<l}(x_k + x_l) = (\prod_{k\neq l}(x_k + x_l))^{\frac{1}{2}}$. In the product $\prod_{k\neq l}(x_k + x_l)$, each of the terms $x_1, \ldots, x_{n-1}$ occurs n times. Therefore, $\prod_{k\neq l}(x_k + x_l) = \prod_{i=1}^{n-1} x_i^n = \gamma^n$. Hence, $\det(V_1) = \gamma^{\frac{n}{2}}$. □

The following lemma shows that the elements in the last column of V_1^{-1}, i.e. $b'_{i,n-1}$s for $i = 0, \ldots, n-1$, are same. The proof technique is similar to [GR13a, Lemma 10].

Lemma 3.11. $b_{i,n-1} = \frac{1}{\gamma}$ *for* $i = 0, \ldots, n-1$.

Proof. Let $i \in \{0, 1, \ldots, n-1\}$ be arbitrary. So, $b_{i,n-1} = \frac{det(V_1')}{\det(V_1)}$, where

$$V_1' = \begin{bmatrix} 1 & 1 & 1 & \ldots & 1 & 1 & \ldots & 1 \\ x_0 & x_1 & x_2 & \ldots & x_{i-1} & x_{i+1} & \ldots & x_{n-1} \\ x_0^2 & x_1^2 & x_2^2 & \ldots & x_{i-1}^2 & x_{i+1}^2 & \ldots & x_{n-1}^2 \\ \vdots & \vdots & \vdots & & \vdots & \vdots & & \\ x_0^{n-2} & x_1^{n-2} & x_2^{n-2} & \ldots & x_{i-1}^{n-2} & x_{i+1}^{n-2} & \ldots & x_{n-1}^{n-2} \end{bmatrix}$$
$$= vand(x_0, x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_{n-1}).$$

Therefore, $\det(V_1') = \prod_{\substack{k<l \\ k,l\neq i}} (x_k + x_l) = (\prod_{\substack{k\neq l \\ k,l\neq i}} (x_k + x_l))^{\frac{1}{2}}$.

Now $\prod_{\substack{k\neq l \\ k,l\neq i}} (x_k + x_l) = \frac{\prod_{k\neq l}(x_k+x_l)}{\prod_{k\neq i}(x_k+x_i)\cdot\prod_{l\neq i}(x_l+x_i)} = \frac{\prod_{k\neq l}(x_k+x_l)}{\prod_{k=1}^{n-1} x_k \cdot \prod_{k=1}^{n-1} x_k} = \frac{\gamma^n}{\gamma^2} = \gamma^{n-2}$.

Therefore, $b_{i,n-1} = \frac{\gamma^{\frac{n-2}{2}}}{\gamma^{\frac{n}{2}}} = \gamma^{-1}$. □

Corollary 3.6. *Let $\{x_0, x_1, \ldots, x_{n-1}\}$ be an additive subgroup G of $\mathbb{F}_{2^r}$ of order n where $x_0 = 0$ and let $y_i = l + x_i$ for $0 \leq i \leq n-1$, where $l \notin G$. Let $V_1 = vand(x_0, x_1, \ldots, x_{n-1})$ and $V_2 = vand(y_0, y_1, \ldots, y_{n-1})$. Then $V_1^{-1}V_2$ is a compact involutory MDS matrix.*

Proof. From Theorem 3.8, we know $V_1^{-1}V_2 = D_1 M D_2$, where D_1 and D_2 are the nonsingular diagonal matrices defined in Theorem 3.8. From Lemma 3.11, we have $D_1 = diag(\frac{1}{\gamma}, \frac{1}{\gamma}, \ldots, \frac{1}{\gamma})$. As the elements x_j form the additive subgroup G, the values $x_i + x_j$, for $i = 0, 1, \ldots, n-1$, includes all the n distinct elements of G for a given j. Consequently, $x_i + y_j = l + x_i + x_j$, for $i = 0, 1, \ldots, n-1$, gives all n distinct elements of $l + G$. Thus, $\prod_{k=0}^{n-1}(x_k + y_0) = \prod_{k=0}^{n-1}(x_k + y_1) = \ldots = \prod_{k=0}^{n-1}(x_k + y_{n-1}) = d$ for some d i.e. $D_2 = diag(d, d, \ldots, d)$. Since M is a compact MDS, $D_1 M D_2$ is compact MDS matrix. Again by Corollary 3.5, $V_1^{-1}V_2$ is involutory. Therefore, $V_1^{-1}V_2$ is compact involutory MDS matrix. □

Note that in [GR13a] it was proved that the constructed MDS matrices from Vandermonde based construction of type 4 are involutory and Hadamard. In the following corollary we prove it in a different way.

Corollary 3.7. *Suppose $\{x_0, x_1, \ldots, x_{n-1}\}$ is an additive subgroup G of $\mathbb{F}_{2^r}$ of order n such that $x_i + x_j = x_{i\oplus j}$ and let $y_i = l + x_i$ for $0 \leq i \leq n-1$, where $l \notin G$. Let $V_1 = vand(x_0, x_1, \ldots, x_{n-1})$ and $V_2 = vand(y_0, y_1, \ldots, y_{n-1})$. Then $V_1^{-1}V_2$ is a Hadamard involutory MDS matrix.*

Proof. As in the proof of Corollary 3.6, we obtain $D_1 = diag(\frac{1}{\gamma}, \frac{1}{\gamma}, \ldots, \frac{1}{\gamma})$ and $D_2 = diag(d, d, \ldots, d)$. Since the constructed matrix M in Cauchy based construction of type 4 is MDS and Hadamard, $D_1 M D_2$ will remain MDS and Hadamard. Again by Corollary 3.5, $V_1^{-1}V_2$ is involutory. Therefore, $V_1^{-1}V_2$ is Hadamard involutory MDS matrix. □

Now we are comparing all the known Vandermonde based constructions with their corresponding Cauchy based constructions in Table 3.1. Let $x_0, x_1, x_2, \ldots, x_{n-1}$ and $y_0, y_1, y_2, \ldots, y_{n-1}$ are $2n$ distinct elements from $\mathbb{F}_{2^r}$ such that $x_i + y_j \neq 0$ for all $0 \leq i, j \leq n-1$. Then the matrices $V_1^{-1}V_2$, $V_2^{-1}V_1$ *and* M are MDS matrices, where $V_1 = vand(x_0, x_1, x_2, \ldots, x_{n-1})$, $V_2 = vand(y_0,\ y_1,\ y_2,\ \ldots,\ y_{n-1})$ and $M = (m_{i,j})$, where $m_{i,j} = \frac{1}{x_i + y_j}$.
Till now, we have discussed Cauchy and Vandermonde based constructions. In these methods the constructed matrices are MDS so they are direct nonrecursive constructions. An objective in designing MDS matrices is to maximize the number of 1's and minimizing the number of distinct elements [JV05b]. The minimum number of distinct elements in Vandermonde and Cauchy based construction is the dimension of the matrix. Next we consider circulant matrix where the number of distinct elements can be even smaller.

3.5 Constructing MDS Matrices from Circulant Matrices and its Variants

To the best of our knowledge, there is currently no known method to provide circulant matrix of arbitrary order which is MDS by construction itself. However, there are constructions of circulant MDS matrices based on search. Though search methods provide efficient MDS matrices of moderate order over moderate size search space, it fails for higher order and large search space. Note, the 4×4 MDS matrix in AES [DR02] has been found by search method. In this section we mainly discuss ideas from [CL19, DR02, GR14, GR15, LS16].

Remark 3.21. *Because of the positional structure of left-circulant matrix, it can be checked that the (i, j)-th entry of the circulant matrix $A = Circ(x_0, \ldots, x_{n-1})$ can be expressed as $(A)_{i,j} = x_{(j-i) mod\ n}$.*

The given lemma highlights a significant property of circulant matrices.

Table 3.1: Comparison between Vandermonde and Cauchy based constructions of MDS matrices over a finite field.

Construction Type	**Vandermonde based Construction** $V_1^{-1}V_2$ and $V_2^{-1}V_1$	**Cauchy based Construction** M
Type 1: No extra condition	1. Need not be involutory 2. Need not be Hadamard 3. Need not be compact	1. Need not be involutory 2. Need not be Hadamard 3. Need not be compact
Type 2: $y_i = l + x_i$, where l is an arbitrary nonzero element in $\mathbb{F}_{2^r}$	1. Involutory and equal 2. Need not be Hadamard 3. Need not be compact	1. Need not be involutory, whereas D_1MD_2 is involutory for some nonsingular diagonal matrices D_1 and D_2 (see Remark 3.20) 2. Need not be Hadamard 3. Need not be compact
Type 3: x_i's are the elements of an additive subgroup $G = \{x_0, x_1, x_2, \ldots, x_{n-1}\}$ of order n of $\mathbb{F}_{2^r}$ and $l \notin G$	1. Involutory and equal 2. Need not be Hadamard 3. compact	1. Need not be involutory, whereas $\frac{1}{c}M$ is involutory, where c is the sum of the elements of any row 2. Need not be Hadamard 3. compact
Type 4: x_i's are the elements of an additive subgroup $G = \{x_0, x_1, x_2, \ldots, x_{n-1}\}$ of order n of $\mathbb{F}_{2^r}$ such that $x_i + x_j = x_{i \oplus j}$ and $l \notin G$	1. Involutory and equal 2. Hadamard 3. compact	1. Need not be involutory, whereas $\frac{1}{c}M$ is involutory, where c is the sum of the elements of any row 2. Hadamard 3. compact

Lemma 3.12. *[RB00] The product of two circulant matrices is also a circulant matrix. Also, the inverse and transpose of a circulant matrix are circulant.*

Circulant matrices can be expressed as polynomials in a suitable permutation matrix. Therefore, we can state the following proposition.

Proposition 3.1. *[RB00, Page 290] A $n \times n$ circulant matrix $A = Circ(x_0, \ldots, x_{n-1})$ can be written in the form $A = x_0I + x_1P + x_2P^2 + \ldots + x_{n-1}P^{n-1}$, where $P = Circ(0, 1, 0, \ldots, 0)$.*

MDS matrices of dimension $2^n \times 2^n$ hold significant cryptographic relevance. It is worth mentioning that in the Advanced Encryption Standard (AES), a $2^2 \times 2^2$ MDS

matrix is utilized. In the MDS-AES scheme proposed by Jorge et al. [NA09], the matrix employed has a dimension of $2^4 \times 2^4$. In Lemma 3.14 it is proved that $2^n \times 2^n$ circulant matrix cannot be both MDS and orthogonal. In Lemma 3.13 and Corollary 3.8, we study two important properties of $2^n \times 2^n$ circulant MDS matrices and using these results we prove Lemma 3.14.

Lemma 3.13. *[GR15, Lemma 4] $Circ(x_0, x_1, \ldots, x_{2^n-1})^{2^n} = (\sum_{i=0}^{2^n-1} x_i^{2^n})I$, where $x_0, \ldots, x_{2^n-1} \in \mathbb{F}_{2^r}$.*

Proof. From Proposition 3.1, $Circ(x_0, x_1, \ldots, x_{2^n-1}) = x_0 I + x_1 P + x_2 P^2 + \ldots + x_{2^n-1} P^{2^n-1}$, where $P = Circ(0, 1, 0, \ldots, 0)$ is a $2^n \times 2^n$ matrix. So,

$$\begin{aligned} Circ(x_0, x_1, \ldots, x_{2^n-1})^{2^n} &= (x_0 I + x_1 P + x_2 P^2 + \ldots + x_{2^n-1} P^{2^n-1})^{2^n} \\ &= x_0^{2^n} I^{2^n} + x_1^{2^n} P^{2^n} + x_2^{2^n} (P^{2^n})^2 + \ldots + x_{2^n-1}^{2^n} (P^{2^n})^{2^n-1} \\ &= (x_0^{2^n} + x_1^{2^n} + x_2^{2^n} + \ldots + x_{2^n-1}^{2^n}) I. \end{aligned}$$

□

Remark 3.22. *If $\sum_{i=0}^{2^n-1} x_i = 1$, then $Circ(x_0, x_1, \ldots, x_{2^n-1})^{2^n} = I$.*

Corollary 3.8. *[GR15, Corollary 1] $\det(Circ(x_0, x_1, \ldots, x_{2^n-1})) = \sum_{i=0}^{2^n-1} x_i^{2^n}$, where $x_0, x_1, \ldots, x_{2^n-1} \in \mathbb{F}_{2^r}$.*

Proof. Let $A = Circ(x_0, x_1, \ldots, x_{2^n-1})$ and $\det(A) = \delta$. So $\delta^{2^n} = (\det(A))^{2^n} = \det(A^{2^n})$. From Lemma 3.13, $A^{2^n} = (\sum_{i=0}^{2^n-1} x_i^{2^n})I$. So, $\delta^{2^n} = \det((\sum_{i=0}^{2^n-1} x_i^{2^n})I) = (\sum_{i=0}^{2^n-1} x_i^{2^n})^{2^n}$. Therefore, $\delta = \sum_{i=0}^{2^n-1} x_i^{2^n}$. □

Lemma 3.14. *[GR15, Lemma 5] For $n \geq 2$, a $2^n \times 2^n$ circulant orthogonal matrix over the finite field $\mathbb{F}_{2^r}$ cannot be an MDS matrix.*

Proof. Consider $A = Circ(a_0, a_1, \ldots, a_{2^n-1})$ to be an orthogonal matrix, where the elements $a_0, a_1, \ldots, a_{2^n-1}$ belong to the finite field $\mathbb{F}_{2^r}$. Let the row vectors of A be denoted as $R_0, R_1, \ldots, R_{2^n-1}$, where $R_0 = (a_0, a_1, \ldots, a_{2^n-1})$, and R_i can be obtained by cyclically rotating R_{i-1} one element to the right. As A is an orthogonal matrix, $R_i \cdot R_j = 0$ for $i \neq j$. Now, let us focus on the cases where $R_0 \cdot R_j = 0$ for $j = 2k+1 : k = 0, \ldots, 2^{n-2} - 1$. This leads to a system of 2^{n-2} equations given by:

$$\sum_{i=0}^{2^n-1} a_i a_{i+1} = 0, \sum_{i=0}^{2^n-1} a_i a_{i+3} = 0, \sum_{i=0}^{2^n-1} a_i a_{i+5} = 0, \ldots, \sum_{i=0}^{2^n-1} a_i a_{i+2^{n-1}-1} = 0,$$

where suffixes are computed modulo 2^n. Adding these equations, we obtain:

$$\sum_{i,j} a_{2i}a_{2j+1} = (a_0 + a_2 + a_4 + \ldots + a_{2^n-2})(a_1 + a_3 + a_5 + \ldots + a_{2^n-1}) = 0. \quad (3.6)$$

It can be observed that matrix A has a $2^{n-1} \times 2^{n-1}$ submatrix $A' = Circ(a_0, a_2, a_4, \ldots, a_{2^n-2})$. This submatrix is formed by selecting rows indexed by $0, 2, 4, \ldots, 2^n - 2$ and columns indexed by $0, 2, 4, \ldots, 2^n - 2$. According to Corollary 3.8, the determinant of this submatrix is given by:

$$\begin{aligned}\det(Circ(a_0, a_2, a_4, \ldots, a_{2^n-2})) &= a_0^{2^{n-1}} + a_2^{2^{n-1}} + a_4^{2^{n-1}} + \ldots + a_{2^n-2}^{2^{n-1}} \\ &= (a_0 + a_2 + a_4 + \ldots + a_{2^n-2})^{2^{n-1}}.\end{aligned}$$

It can also be checked that A has another $2^{n-1} \times 2^{n-1}$ submatrix $A'' = Circ(a_1, a_3, a_5, \ldots, a_{2^n-1})$ which is formed by selecting rows indexed by $0, 2, 4, \ldots, 2^n - 2$ and columns indexed by $1, 3, 5, \ldots, 2^n - 1$. The determinant of this submatrix is given by:

$$\begin{aligned}\det(Circ(a_1, a_3, a_5, \ldots, a_{2^n-1})) &= a_1^{2^{n-1}} + a_3^{2^{n-1}} + a_5^{2^{n-1}} + \ldots + a_{2^n-1}^{2^{n-1}} \\ &= (a_1 + a_3 + a_5 + \ldots + a_{2^n-1})^{2^{n-1}}.\end{aligned}$$

Now, based on Equation 3.6, it can be concluded that at least one of the submatrices A' and A'' is singular. Therefore, it can be deduced that the matrix A is not an MDS matrix. □

Remark 3.23. *Lemma 3.14 is a slightly modified version of [GR15, Lemma 5]. They did not mention that n should be ≥ 2. We observe that the result is not true for the matrices of order 2 . For example, consider the matrix $A = \begin{bmatrix} \alpha & 1+\alpha \\ 1+\alpha & \alpha \end{bmatrix}$, where α is a primitive element of $\mathbb{F}_{2^4}$ whose constructing polynomial is $x^4 + x + 1$. It can be easily verified that the matrix A is both circulant MDS and orthogonal. With the above example it may be checked that similar errors were present in the original version of Lemma 3.17, Theorem 3.9, Theorem 3.10 Theorem 3.11.*

Remark 3.24. *While the circulant MDS matrices of size $2^n \times 2^n$ are not orthogonal, it is possible for circulant MDS matrices of other orders to be orthogonal. For instance, consider the 3×3 matrix $A' = Circ(\alpha, 1+\alpha^2+\alpha^3+\alpha^4+\alpha^6, \alpha+\alpha^2+\alpha^3+\alpha^4+\alpha^6)$ and the 6×6 matrix $A'' = Circ(1, 1, \alpha, 1+\alpha^2+\alpha^3+\alpha^5+\alpha^6+\alpha^7, \alpha+\alpha^5, \alpha^2+\alpha^3+\alpha^6+\alpha^7)$, where α is a root of the polynomial $x^8 + x^4 + x^3 + x + 1$, which is the constructing polynomial of $\mathbb{F}_{2^8}$. It can be verified that both A' and A'' are orthogonal MDS matrices.*

In Lemma 3.17 we show that circulant matrices cannot be both involutory and MDS. Prior to that, we analyze two beneficial properties in Lemma 3.15 and Lemma 3.16.

Lemma 3.15. *[GR15, Lemma 7] Let $A = Circ(x_0, x_1, \ldots, x_{2n-1})$ be a $2n \times 2n$ circulant matrix, where $x_0, \ldots, x_{2n-1} \in \mathbb{F}_{2^r}$. Then*

$$A^2 = Circ(x_0^2 + x_n^2, 0, x_1^2 + x_{n+1}^2, 0, \ldots, x_{n-1}^2 + x_{2n-1}^2, 0).$$

Proof. From Proposition 3.1,

$$A = x_0 I + x_1 P + x_2 P^2 + \ldots + x_{2n-1} P^{2n-1},$$

where P is a $2n \times 2n$ matrix given by $P = Circ(0, 1, 0, \ldots, 0)$. So

$$\begin{aligned}
A^2 &= x_0^2 I + x_1^2 P^2 + x_2^2 P^4 + \ldots + x_{2n-1}^2 P^{2(2n-1)} \\
&= (x_0^2 I + x_n^2 P^{2n}) + (x_1^2 P^2 + x_{n+1}^2 P^{2n+2}) + \ldots + (x_{n-1}^2 P^{2(d-1)} + x_{2n-1}^2 P^{2(2n-1)}) \\
&= (x_0^2 + x_n^2) I + (x_1^2 + x_{n+1}^2) P^2 + \ldots + (x_{n-1}^2 + x_{2n-1}^2) P^{(2n-2)} \\
&= Circ(x_0^2 + x_n^2, 0, x_1^2 + x_{n+1}^2, 0, \ldots, x_{n-1}^2 + x_{2n-1}^2, 0).
\end{aligned}$$

□

Lemma 3.16. *[GR15, Lemma 8] Let $A = Circ(x_0, x_1, \ldots, x_{2n})$ be a $(2n+1) \times (2n+1)$ circulant matrix, where $x_0, \ldots, x_{2n} \in \mathbb{F}_{2^r}$. Then*

$$A^2 = Circ(x_0^2, x_{n+1}^2, x_1^2, x_{n+2}^2, \ldots, x_{n-1}^2, x_{2n}^2, x_n^2).$$

Proof. From Proposition 3.1, A can be written as

$$A = x_0 I + x_1 P + x_2 P^2 + \ldots + x_{2n} P^{2n},$$

where $P = Circ(0, 1, 0, \ldots, 0)$ is a $(2n+1) \times (2n+1)$ matrix. So

$$\begin{aligned}
A^2 &= x_0^2 I + x_1^2 P^2 + x_2^2 P^4 + \ldots + x_{2n}^2 P^{2(2n)} \\
&= x_0^2 I + x_{n+1}^2 P^{(2n+1+1)} + x_1^2 P^2 + x_{n+2}^2 P^{(2n+1+3)} + x_2^2 P^4 + \ldots \\
&\quad + x_{n-1}^2 P^{2n-2} + x_{2n}^2 P^{(2n+1+2n-1)} + x_n^2 P^{2n} \\
&= x_0^2 I + x_{n+1}^2 P + x_1^2 P^2 + x_{n+2}^2 P^3 + \ldots + x_{n-1}^2 P^{(2n-2)} + x_{2n}^2 P^{(2n-1)} + x_n^2 P^{2n} \\
&= Circ(x_0^2, x_{n+1}^2, x_1^2, x_{n+2}^2, \ldots, x_{n-1}^2, x_{2n}^2, x_n^2).
\end{aligned}$$

□

Remark 3.25. *If we consider the matrix $A = Circ(a_0, \ldots, a_{2^n-1})$ with $\sum_{i=0}^{2^n-1} a_i = 1$, then, according to Remark 3.22, we have $A^{2^n} = I$. Therefore, the inverse of A can be computed as $A^{-1} = A^{2^n-1} = \prod_{k=0}^{n-1} A^{2^k}$. It is worth noting that matrices of the form A^{2^k} for $k > 0$ are computationally efficient, as they contain mostly zero elements.*

Consequently, the InvMixColumn operation can be implemented by preprocessing the multiplication of the input matrix by $A^2 \times A^4 \times \ldots \times A^{2^{n-1}}$, followed by the MixColumn step. For instance, in the case of $n = 2$, the inverse of A can be computed as $A^{-1} = A \times A^2$.

Remark 3.26. *In AES [DR02], the MixColumn operation utilizes an MDS matrix $M = Circ(\alpha, 1+\alpha, 1, 1)$, where α is a root of the polynomial $x^8 + x^4 + x^3 + x + 1$. Barreto observed that in the* InvMixColumn *operation [DR02] of decryption, using $M \times M^2 = Circ(\alpha, 1+\alpha, 1, 1) \times Circ(1+\alpha^2, 0, \alpha^2, 0)$ instead of M^{-1} can lead to a more efficient implementation. This observation is a consequence of Lemma 3.13, Lemma 3.15, and Remark 3.25.*

Lemma 3.17. *[GR15, Lemma 9] Circulant involutory matrices of order $n \geq 3$ over $\mathbb{F}_{2^r}$ are not an MDS matrix.*

Proof. Consider a $2n \times 2n$ involutory circulant matrix $A = Circ(x_0, x_1, \ldots, x_{2n-1})$. It follows that $A^2 = I$. However, applying Lemma 3.15, we have $A^2 = Circ(x_0^2 + x_n^2, 0, x_1^2 + x_{n+1}^2, 0, \ldots, x_{n-1}^2 + x_{2n-1}^2, 0)$. From this, we can observe that $x_1^2 + x_{n+1}^2 = 0$. Also, by considering the 2×2 submatrix $Circ(x_1, x_{n+1})$ of A, obtained from the 0-th and n-th rows and the 1-st and $(n+1)$-th columns, we find that $\det(Circ(x_1, x_{n+1})) = x_1^2 + x_{n+1}^2 = 0$. Consequently, A is not an MDS matrix.

Similarly, for the $(2n+1) \times (2n+1)$ involutory circulant matrix $A = Circ(x_0, x_1, \ldots, x_{2n})$, using Lemma 3.16, we find that $A^2 = Circ(x_0^2, x_{n+1}^2, x_1^2, x_{n+2}^2, \ldots, x_{2n}^2, x_n^2)$. However, since A is involutory, we have $A^2 = I$. From this, it becomes evident that $x_i = 0$ for all $i \in \{1, 2, \ldots, 2n\}$. Therefore, matrix A is not an MDS matrix. □

Remark 3.27. *Over a field of odd characteristic, even order involutory circulant matrix is not MDS whereas odd order involutory circulant matrix may be MDS [CL19].*

In [GR15], it was established that Type-I circulant-like MDS matrices of even order cannot be involutory or orthogonal. However, the discussion did not cover the case of matrices with odd order. In this section, we aim to prove that Type-I circulant-like MDS matrices of odd order also cannot be involutory or orthogonal.

Lemma 3.18. *[GR15, Lemma 6] A Type-I circulant-like MDS matrix of size $2n \times 2n$ over $\mathbb{F}_{2^r}$ is not orthogonal.*

Proof. Let $M = \begin{bmatrix} a & \mathbf{1} \\ \mathbf{1}^T & A \end{bmatrix}$, where $A = Circ(1, x_1, \ldots, x_{2n-2})$. Now $M \times M^T = \begin{bmatrix} a^2+1 & \mathbf{c} \\ \mathbf{c}^T & B \end{bmatrix}$, where $\mathbf{c} = \underbrace{(c, \ldots, c)}_{\text{2n-1 times}}$, $c = a + 1 + \sum_{i=1}^{2n-2} x_i$, $B = U + A \times A^T$ and

$U = (u_{i,j})$, where $u_{i,j} = 1$ for $0 \leq i, j \leq 2n-2$. Suppose M is orthogonal. Then we have $M \times M^T = I$. This implies that $a^2 + 1 = 1$, which leads to the conclusion that $a = 0$. Therefore, we can deduce that M is not an MDS matrix. □

In the following lemma we prove that there is no orthogonal *Type-I circulant-like matrix* of odd order as well.

Lemma 3.19. *A Type-I circulant-like matrix of size* $(2n+1) \times (2n+1)$ *over* $\mathbb{F}_{2^r}$ *is not orthogonal.*

Proof. Let $M = \begin{bmatrix} a & \mathbf{1} \\ \mathbf{1}^T & A \end{bmatrix}$, where $A = Circ(1, x_1, \ldots, x_{2n-1})$. Now $M \times M^T = \begin{bmatrix} a^2 & \mathbf{c} \\ \mathbf{c}^T & B \end{bmatrix}$, where $\mathbf{c} = \underbrace{(c, \ldots, c)}_{\text{2n times}}$, $c = a + 1 + \sum_{i=1}^{2n-1} x_i$, $B = U + A \times A^T$ and $U = (u_{i,j})$ is a $2n \times 2n$ matrix with $u_{i,j} = 1$. Suppose M is orthogonal, $M \times M^T = I$, which gives $a^2 = 1$ and hence $a = 1$.
Now $c = a + 1 + \sum_{i=1}^{2n-1} x_i = \sum_{i=1}^{2n-1} x_i$. Again as M is orthogonal, we have $c = 0$ which implies that $\sum_{i=1}^{2n-1} x_i = 0$. Therefore, $\sum_{i=1}^{2n-1} x_i^2 = 0 \implies 1 + \sum_{i=1}^{2n-1} x_i^2 = 1$.
So, $(AA^T)_{i,i} = 1 + \sum_{i=1}^{2n-1} x_i^2 = 1$. Thus, $(MM^T)_{1,1} = (B)_{0,0} = (U)_{0,0} + (AA^T)_{0,0} = 1 + 1 = 0$, Which is a contradiction. Hence, M cannot be orthogonal. □

In [GR15], it was proved that there is no involutory *Type-I circulant-like matrix* of even order. Therefore, we have the following lemma from [GR15].

Lemma 3.20. *[GR15, Lemma 10] A Type-I circulant-like matrix of size* $2n \times 2n$ *over* $\mathbb{F}_{2^r}$ *cannot be involutory.*

Proof. Let $M = \begin{bmatrix} a & \mathbf{1} \\ \mathbf{1}^T & A \end{bmatrix}$, where $A = Circ(1, x_1, \ldots, x_{2n-2})$. Now, $M^2 = \begin{bmatrix} a^2 + 1 & \mathbf{c} \\ \mathbf{c}^T & B \end{bmatrix}$, where $\mathbf{c} = \underbrace{(c, \ldots, c)}_{\text{2n-1 times}}$, $c = a + 1 + \sum_{i=1}^{2n-2} x_i$, $B = U + A^2$ and $U = (u_{i,j})$ is the $(2n-1) \times (2n-1)$ matrix with $u_{i,j} = 1$. $(A^2)_{0,0} = 1$, consequently $(M^2)_{1,1} = (B)_{0,0} = (U)_{0,0} + (A^2)_{0,0} = 1 + 1 = 0$ and so $M^2 \neq I$. Hence, we can conclude that M cannot be involutory. □

Remark 3.28. *In Lemma 3.20, if* $n = 2$ *and* $A = Circ(1, b, a)$,

$$M^2 = \begin{bmatrix} a^2+1 & (b+1) & (b+1) & (b+1) \\ (b+1) & 0 & (1+a^2) & (1+b^2) \\ (b+1) & (1+b^2) & 0 & (1+a^2) \\ (b+1) & (1+a^2) & (1+b^2) & 0 \end{bmatrix} \neq I.$$

Hence, the matrix M is not involutory. In the case where $a = \alpha$ and $b = 1 + \alpha^{-1}$, where α is a root of the constructing polynomial $x^8 + x^7 + x^6 + x^5 + x^4 + x^3 + 1$ of $\mathbb{F}_{2^8}$, we obtain the matrix M which is utilized in the block cipher FOX64 [JV05a].

In the following lemma we also prove that there is no involutory *Type-I circulant-like matrix* of odd order.

Lemma 3.21. *A Type-I circulant-like matrix of size $(2n+1) \times (2n+1)$ over $\mathbb{F}_{2^r}$ cannot be involutory.*

Proof. Let $M = \begin{bmatrix} a & \mathbf{1} \\ \mathbf{1}^T & A \end{bmatrix}$, where $A = Circ(1, x_1, \ldots, x_{2n-1})$. Now, $M^2 = \begin{bmatrix} a^2 & \mathbf{c} \\ \mathbf{c}^T & B \end{bmatrix}$, where $\mathbf{c} = \underbrace{(c, \ldots, c)}_{\text{2n times}}$, $c = a + 1 + \sum_{i=1}^{2n-1} x_i$, $B = U + A^2$ and $U = (u_{i,j})$ is the $2n \times 2n$ matrix with $u_{i,j} = 1$. Since A is even circulant matrix by Lemma 3.15, we have $(A^2)_{0,1} = 0$. Therefore, $(M^2)_{1,2} = (B)_{0,1} = (U)_{0,1} + (A^2)_{0,1} = 1$. Thus, $M^2 \neq I$. Hence, M is not involutory. □

In order to explore the construction of involutory MDS matrices using *Type-II circulant-like matrices*, Gupta et al. [GR15] demonstrated that *Type-II circulant-like matrices* are inherently involutory.

Lemma 3.22. *[GR15, Lemma 11] Over $\mathbb{F}_{2^r}$, Type-II circulant-like matrices are involutory.*

Proof. Consider a $n \times n$ circulant matrix A and $M = \begin{bmatrix} A & A^{-1} \\ A^3 + A & A \end{bmatrix}$ be a *Type-II circulant-like matrix*. Now,

$$M^2 = \begin{bmatrix} A^2 + A^{-1}(A^3 + A) & AA^{-1} + A^{-1}A \\ (A^3 + A)A + A(A^3 + A) & (A^3 + A)A^{-1} + A^2 \end{bmatrix}$$
$$= \begin{bmatrix} A^2 + A^2 + I_{n\times n} & \mathbf{0} \\ \mathbf{0} & A^2 + I_{n\times n} + A^2 \end{bmatrix} = I_{2n\times 2n}.$$

Hence, M is involutory. □

Example 3.8. *Consider the Type-II circulant-like matrix $A = TypeII(Circ(\alpha, 1, 1+\alpha^2))$ over $\mathbb{F}_{2^8}$ with constructing polynomial is $x^8 + x^4 + x^3 + x + 1$, where α is a root of the constructing polynomial. Then A is an involutory MDS matrix.*

In [YMT97], the authors explored the construction of $2n \times 2n$ MDS matrices using a random $n \times n$ MDS matrix as a submatrix. However, their attempts to find an MDS matrix through random search for $n = 4$ were unsuccessful. In a separate study [GR15], it was proven that if the $n \times n$ submatrix is a circulant MDS matrix and n is even, then the corresponding $2n \times 2n$ matrix is not an MDS matrix. Based on this result, we state the following lemma from [GR15].

Lemma 3.23. *[GR15, Lemma 12] For even values of n, any $2n \times 2n$ Type-II circulant-like matrix over $\mathbb{F}_{2^r}$ is not an MDS matrix.*

Remark 3.29. *It is important to highlight that circulant and Type-I circulant MDS matrices do not possess the properties of being involutory and orthogonal. Therefore, our attention is directed towards constructing matrices that exhibit these properties while still allowing for efficient implementation of their inverses. It should be noted that Type-II circulant-like matrices are always involutory; however, when the dimensions are in the form of $2(2n) \times 2(2n)$, they are not MDS. For further information, we refer to Table 3.2.*

Now we briefly discuss left-circulant matrices for the construction of MDS matrices.

Remark 3.30. *Because of the positional structure of left-circulant matrix, it can be checked that the (i, j)-th entry of the left-circulant matrix $A = l\text{-}Circ(x_0, \ldots, x_{n-1})$ can be expressed as $(A)_{i,j} = x_{(i+j) mod\ n}$.*

Many properties of left-circulant matrices are similar to circulant matrices, in this context we provide a few properties through Proposition 3.2 and Proposition 3.3. In [LS16], these propositions were used to prove Theorem 3.9, but here we provide an alternative proof.

Proposition 3.2. *[LS16, Proposition 4] The multiplication of two left-circulant matrices results in a circulant matrix.*

Proof. Let $A = l\text{-}Circ(x_0, x_1, ..., x_{n-1})$ and $B = l\text{-}Circ(y_0, y_1, ..., y_{n-1})$ be two left-circulant matrices. Then the (i, j)-th entry of their product is

$$\sum_{k=0}^{n-1} (A)_{i,k} \cdot (B)_{k,j} = \sum_{k=0}^{n-1} x_{i+k} \cdot y_{k+j} = \sum_{k=0}^{n-1} x_k \cdot y_{k+(j-i)}.$$

This shows that AB is a circulant matrix. □

Proposition 3.3. *[LS16, Proposition 5] For $2^n \times 2^n$ left-circulant matrix $L = l\text{-}Circ(x_0, x_1, ..., x_{2^n-1})$ over $\mathbb{F}_{2^r}$, $L^{2^{n+1}} = (\sum_{i=0}^{2^n-1} x_i)^{2^{n+1}} I$ and $\det(L) = (\sum_{i=0}^{2^n-1} x_i)^{2^n}$.*

Proof. By Proposition 3.2, L^2 is circulant with (i,j)-th entry $\sum_{k=0}^{2^n-1} x_k \cdot x_{k+(j-i)}$ and hence

$$(L^2)^{2^n} = (\sum_{i=0}^{2^n-1} \sum_{k=0}^{2^n-1} x_k \cdot x_{k+i})^{2^n} I = ((\sum_{k=0}^{2^n-1} x_k)^2)^{2^n} I = (\sum_{k=0}^{2^n-1} x_k)^{2^{n+1}} I,$$

which also implies that $\det(L) = (\sum_{i=0}^{2^n-1} x_k)^{2^n}$. □

Remark 3.31. *From Lemma 2.11, we know that if A is MDS matrix then for any permutation matrix P, PA is also MDS matrix. Also from Remark 3.24, we know that there may exists circulant MDS matrix A= $Circ(x_0, x_1, ..., x_{n-1})$ over $\mathbb{F}_{2^r}$ which is orthogonal, where n is not the power of 2. Considering Remark 2.12, we establish that PA = l-$Circ(x_0, x_1, \ldots, x_{n-1})$, where P corresponds to the permutation matrix mentioned in Remark 2.12. Also, MDS property and orthogonality of A will not be disturbed by pre-multiplying with P. Consequently, the matrix PA =l-$Circ(x_0, x_1, ..., x_{n-1})$ is an orthogonal MDS matrix and, thus, an involutory MDS matrix.*

In [GR15], it was demonstrated that a circulant matrix cannot possess both involutory and MDS properties. Additionally, it was shown that a $2^n \times 2^n$ circulant matrix cannot cannot be both MDS and orthogonal. Similarly, in [LS16, Theorem 4], it was proven that a $2^n \times 2^n$ left-circulant matrix cannot simultaneously be MDS and involutory (orthogonal). In the subsequent theorem, we present an alternative and simpler proof of Theorem 4 in [LS16]. For the original proof, readers are advised to go through [LS16].

Theorem 3.9. *[LS16, Theorem 4] For $n \geq 2$, if L is a $2^n \times 2^n$ left-circulant MDS matrix over $\mathbb{F}_{2^r}$, then L is not involutory (orthogonal).*

Proof. Assume that L= l-$Circ(x_0, x_1, ..., x_{2^n-1})$ is an involutory MDS matrix over $\mathbb{F}_{2^r}$. As L is symmetric it is also an orthogonal MDS matrix. It is easy to check that PL= $Circ(x_0, x_1, ..., x_{2^n-1})$, where P is the permutation matrix as in Remark 3.31. Since P is a permutation matrix, the matrix PL = $Circ(x_0, x_1, ..., x_{2^n-1})$ will also be orthogonal and MDS. By Lemma 3.14 it is a contradiction. □

Remark 3.32. *While the $2^n \times 2^n$ left-circulant MDS matrices are not involutory, it is possible for left-circulant MDS matrices of other orders to exhibit involutory property. For instance, over $\mathbb{F}_{2^8}$ with constructing polynomial $x^8 + x^6 + x^5 + x^2 + 1$, the 5×5 matrix l-$Circ(1, \alpha, \alpha^7 + \alpha^5 + \alpha^4 + \alpha + 1, \alpha^7 + \alpha^5 + \alpha^4 + \alpha^3 + \alpha + 1, \alpha^3 + \alpha)$ and the*

6×6 *matrix* l*-Circ*$(1, 1, \alpha^7 + \alpha^5 + \alpha^4 + \alpha + 1, \alpha^5 + \alpha^3 + \alpha^2, \alpha^2, \alpha^7 + \alpha^4 + \alpha^3 + \alpha)$ *are MDS and involutory, where* $\alpha^8 + \alpha^6 + \alpha^5 + \alpha^2 + 1 = 0$.

We will finish this section by discussing an equivalence relation between circulant matrices. In [LS16], authors provided an equivalence relation by which we can partition the $n!$ possible circulant matrices of order n into $\frac{(n-1)!}{\phi(n)}$ equivalence classes each containing $n\phi(n)$ circulant matrices having the same branch number. Here ϕ is the Euler's totient function.

In [LS16], Liu and Sim provided a necessary and sufficient condition for two circulant matrices to be permutation equivalent. In this chapter, we record the lemma without its original proof. We will provide an alternative proof for this which is more basic. For the original proof, reader are requested to go through Lemma 1 of [LS16].

Lemma 3.24. *[LS16, Lemma 1] Given two circulant matrices* $C = Circ(x_0, x_1, \ldots, x_{n-1})$ *and* $C^\sigma = Circ(x_{\sigma(0)}, x_{\sigma(1)}, \ldots, x_{\sigma(n-1)})$*, we have* $C \sim C^\sigma$*, where* $\sim$ *is the equivalence relation defined in Definition 2.20, if and only if* σ *is an index permutation that satisfies* $\sigma(i) = (bi + a) \bmod n$*,* $\forall i \in \{0, 1, \ldots, n-1\}$*, with* $a, b \in \mathbb{Z}_n$ *and* $gcd(b, n) = 1$.

Proof. If part: Suppose that $\sigma(i) = bi + a$ such that $gcd(b, n) = 1$. We have to show that for some permutation matrices P and Q, $PCQ = C^\sigma$. We will construct three permutation matrix P_1, P_2, Q_1 such that $C^\sigma = P_1P_2CQ_1$. Note that the inverse of a permutation matrix Q is Q^T and product of two permutation matrices is a permutation matrix. Construct the permutation matrix Q_1 whose i-th column has 1 at $i.b$-th position for $0 \leq i \leq n-1$. Let $P_1 = Q_1^T = Q_1^{-1}$. Note that Q_1 is a permutation matrix as $gcd(b, n) = 1$. Now

$$\begin{aligned} P_1CQ_1 &= c_0P_1Q_1 + c_1P_1PQ_1 + c_2P_1P^2Q_+ \ldots + c_{n-1}P_1P^{n-1}Q_1 \\ &= c_0Q_1^TQ_1 + c_1Q_1^TPQ_1 + c_2Q_1^TP^2Q_1 + \ldots + c_{n-1}Q_1^TP^{n-1}Q_1 \\ &= c_0I + c_1Q_1^TPQ_1 + c_2(Q_1^TPQ_1)^2 + \ldots + c_{n-1}(Q_1^TPQ_1)^{n-1}. \end{aligned}$$

Since Q_1 and Q_1^T are two permutation matrices, it is easy to check that $Q_1^TPQ_1 = P^j$ for some j. Therefore,

$$P_1CQ_1 = c_0I + c_1P^j + c_2P^{2j} + \ldots + c_{n-1}P^{(n-1)j}.$$

So P_1CQ_1 is a circulant matrix.

In another direction it is easy to check that the first row of P_1CQ_1 =(c_0, c_b, c_{2b},

$\ldots,\ c_{(n-1)b})$. Now consider the permutation matrix P_2 =Circ(0, 0, $\ldots$, $\underbrace{1}_{\text{(n-a)-th position}}$, $\ldots$, 0). It is easy to check that $P_2C = Circ(c_a,\ c_{a+1},\ c_{a+2},\ \ldots,\ c_{a+n-1})$ and $P_1P_2CQ_1 = Circ(c_a, c_{a+b}, c_{a+2b}, \ldots, c_{a+(n-1)b})$.

Only if part: Suppose that C and C^σ are two circulant matrices such that $C \sim C^\sigma$. Therefore, there exists two permutation matrix Q_1 and Q_2 such that $C^\sigma = Q_1CQ_2$. Since C is a circulant matrix, by Proposition 3.1 we have

$$C = c_0I + c_1P + c_2P^2 + \ldots + c_{n-1}P^{n-1},$$

where $(c_0, c_1, c_2, \ldots, c_{n-1})$ is the first row of C. Therefore,

$$Q_1CQ_2 = Q_1c_0Q_2 + Q_1c_1PQ_2 + Q_1c_2P^2Q_2 + \ldots + Q_1c_{n-1}P^{n-1}Q_2. \tag{3.7}$$

Comparing the positions of c_i in both L.H.S. and R.H.S. of Equation 3.7 and since Q_1CQ_2 is circulant, $Q_1c_iP^iQ_2 = c_iP^{d_i}$ for some $d_i \geq 0$. So from $Q_1c_0Q_2 = c_0P^{d_0}$, we have

$$Q_1Q_2 = P^{d_0} \implies Q_1 = P^{d_0}Q_2^{-1}.$$

Again

$$Q_1PQ_2 = P^{d_1} \implies P^{d_0}Q_2^{-1}PQ_2 = P^{d_1} \implies Q_2^{-1}PQ_2 = P^{d_1-d_0}.$$

Let $b = (d_1 - d_0) \bmod n$, so $Q_2^{-1}PQ_2 = P^b$. Therefore,

$$\begin{aligned} Q_1P^iQ_2 &= P^{d_0}Q_2^{-1}P^iQ_2 \\ &= P^{d_0}(Q_2^{-1}PQ_2)^i \\ &= P^{d_0}P^{bi} = P^{d_0+bi}. \end{aligned}$$

Therefore, $C^\sigma = Q_1CQ_2 = \sum_{i=0}^{n-1} c_iP^{d_0+bi} \implies \sigma(i) = bi + d_0$. Take $d_0 = a$, we have $\sigma(i) = bi + a$. As σ is a permutation on $\{0, 1, \ldots, n-1\}$, we must have $gcd(b, n) = 1$. □

Remark 3.33. *In an equivalence class defined by the relation $\sim$ all the circulant matrices have same branch number. But it may be noted that different equivalence classes can have same branch number. In [LS16], authors proved that these equivalence classes, defined by the relation $\sim$, represent the most compact form for circulant matrices in terms of their equivalence classes. It seems interesting, but difficult to define an equivalence class such that different equivalence classes will have different*

branch numbers and all MDS circulant matrices will be in one equivalence class.

3.6 Constructing MDS Matrices from Toeplitz and Hankel Matrices

Toeplitz matrices are used to construct MDS matrices using search technique. It has similarity to the constructions of circulant MDS matrices discussed in [GR15]. In this section, we mainly discuss the results from [SS16, SS17].

Theorem 3.10. *[SS16, Theorem 1] Toeplitz matrices of order $n \geq 3$ over $\mathbb{F}_{2^r}$ cannot be both MDS and involutory.*

Proof. Let $A = Toep(a_0, a_1, \ldots, a_{n-1};\ a_{-1}, a_{-2}, \ldots, a_{-(n-1)})$ be a $n \times n$ Toeplitz matrix which is both involutory and MDS, where $n \geq 3$. It can be checked that the (i, j)-th entry of A can be expressed as $(A)_{i,j} = a_{(j-i)}$.

Case 1. When n is odd.
The $(n-2)$-th element in the 0-th row of A^2 is

$$\begin{aligned}(A^2)_{0,n-2} &= A_{row(0)} \cdot A_{column(n-2)} \\ &= a_0a_{n-2} + a_1a_{n-3} + \ldots + a_{\frac{n-1}{2}}a_{\frac{n-3}{2}} + \ldots + a_{n-2}a_0 + a_{n-1}a_{-1} \\ &= a_{n-1}a_{-1}.\end{aligned}$$

Since A is involutory then

$$a_{n-1}a_{-1} = 0,$$

which implies that $a_{n-1} = 0$ or $a_{-1} = 0$. This contradicts that A is MDS.
Case 2. When n is even.

$$\begin{aligned}(A^2)_{0,n-2} &= A_{row(0)} \cdot A_{column(n-2)} \\ &= a_0a_{n-2} + a_1a_{n-3} + \ldots + a_{\frac{n-2}{2}}a_{\frac{n-2}{2}} + \ldots + a_{n-2}a_0 + a_{n-1}a_{-1} \\ &= a_{\frac{n-2}{2}}^2 + a_{n-1}a_{-1}.\end{aligned}$$

Therefore, as A is an involution, we have

$$a_{\frac{n-2}{2}}^2 + a_{n-1}a_{-1} = 0.$$

Consider the 2×2 submatrix of A formed by the 0-th and $\frac{n}{2}$-th row and $\frac{n-2}{2}$-th and

$(n-1)$-th column,

$$T = \begin{bmatrix} a_{\frac{n-2}{2}} & a_{n-1} \\ a_{-1} & a_{\frac{n-2}{2}} \end{bmatrix}$$

which is singular. Therefore, A is not MDS. □

Like circulant matrices, Toeplitz matrices of order 2^n cannot be both orthogonal and MDS.

Theorem 3.11. *[SS16, Theorem 2] For $n \geq 2$, any $2^n \times 2^n$ Toeplitz orthogonal matrix over $\mathbb{F}_{2^r}$ is not an MDS matrix.*

Proof. Suppose that $A = Toep(a_0, a_1, \ldots, a_{2^n-1};\ a_{-1}, a_{-2}, \ldots, a_{-(2^n-1)})$ be a Toeplitz matrix of order 2^n which is both orthogonal and MDS. It can be checked that the (i,j)-th entry of A can be expressed as $(A)_{i,j} = a_{(j-i)}$. Let δ_i be the diagonal element of AA^T for $i = 0, 1, \ldots, 2^n - 1$. Then

$$\delta_i = \sum_{j=0}^{2^n-1} a_{j-i}^2 = 1 \text{ for } \ i = 0, 1, \ldots, 2^n - 1.$$

Considering the pair of equations (δ_i and δ_{i+1}), we get

$$a_{-i} = a_{2^n-i} \text{ for } \ i = 0, 1, \ldots, 2^n - 1.$$

Therefore, A is indeed a circulant matrix. Therefore, from Lemma 3.14, A cannot be MDS. □

Remark 3.34. *Circulant matrices are a particular type of Toeplitz matrices, and thus, from Remark 3.24, we can say that Toeplitz orthogonal MDS matrices of orders other than 2^n may exist over $\mathbb{F}_{2^r}$.*

Now we introduce Hankel matrices which are closely related to the Toeplitz matrices in which each ascending skew diagonal from left to right is constant.

Remark 3.35. *From Lemma 2.11, we know that if T is MDS matrix then for any permutation matrix P, PT is also MDS matrix. Also, from Remark 3.34, we know that there may exists Toeplitz MDS matrix $T = Toep(a_0, a_1, \ldots, a_{n-2}, a_{n-1};\ a_{-1}, a_{-2}, \ldots, a_{-(n-1)})$ over $\mathbb{F}_{2^r}$ which is orthogonal, where n is not a power of 2. Consider the permutation matrix*

$$P = \begin{bmatrix} 0 & 0 & 0 & \dots & 0 & 0 & 1 \\ 0 & 0 & 0 & \dots & 0 & 1 & 0 \\ 0 & 0 & 0 & \dots & 1 & 0 & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots & \vdots & \vdots \\ 1 & 0 & 0 & \dots & 0 & 0 & 0 \end{bmatrix}.$$

Now it is easy to check that $PT = H$*, where* $H = Hank(a_{-(n-1)}, a_{-(n-2)}, \dots, a_{-1}, a_0; a_1, a_2, \dots, a_{n-1})$*. But MDS property and orthogonality of* T *will not be disturbed by the multiplication with* P*. Therefore, the matrix* $H = Hank(a_{-(n-1)}, a_{-(n-2)}, \dots, a_{-1}, a_0; a_1, a_2, \dots, a_{n-1})$ *will be an orthogonal MDS matrix and so an involutory MDS matrix.*

It is established that a Toeplitz matrix of order $n \geq 3$ is not an involutory MDS and an orthogonal Toeplitz matrix of size $2^n \times 2^n$ is not an MDS. Similarly, we demonstrate that an involutory (orthogonal) Hankel matrix of size $2^n \times 2^n$ is also not an MDS. Similar results can be found in Remark 3.31 and Theorem 3.9, which highlight analogous results between circulant and left-circulant matrices.

Theorem 3.12. *For* $n \geq 2$*, if* H *is a* $2^n \times 2^n$ *Hankel MDS matrix over* $\mathbb{F}_{2^r}$*, then* H *is not involutory (orthogonal).*

Proof. Let $H = Hank(a_0, a_1, \dots, a_{2^n-1}; a_{2^n}, a_{2^n+1}, \dots, a_{2^{n+1}-2})$ be a $2^n \times 2^n$ Hankel matrix. It can be checked that $(H)_{i,j} = a_{i+j}$. Assume that H is an involutory MDS matrix over $\mathbb{F}_{2^r}$. Therefore, H is an orthogonal MDS matrix. It is easy to check that $PH = T$, where P is the permutation matrix as defined in Remark 3.35 and $T = Toep(a_{2^n-1}, a_{2^n}, a_{2^n+1}, \dots, a_{2^{n+1}-2}; a_{2^n-2}, a_{2^n-3}, \dots, a_1, a_0)$. Since P is a permutation matrix and H is an orthogonal MDS matrix, $T = PH$ is an orthogonal MDS matrix. Therefore, T is an orthogonal MDS matrix of order 2^n, which is a contradiction by Theorem 3.11. Hence, H cannot be an involutory. □

Remark 3.36. *Left-circulant matrices are a particular type of Hankel matrices, and thus, from Remark 3.32, we can say that Hankel involutory MDS matrices of orders other than* 2^n *may exist over* $\mathbb{F}_{2^r}$*.*

Obtaining involutory MDS matrices is of particular interest. However, circulant and Toeplitz matrices cannot generate involutory MDS matrices, as demonstrated in Lemma 3.17 and Theorem 3.10, respectively. On the other hand, left-circulant matrices, Hankel matrices, Hadamard matrices, GHadamard matrices [PSA+18, PSAS22, TSP+23], and subfield construction methods [Ota22, PSAS23, SKOP15] have the

capability to do so. Apart from these specialized matrix types and methods, an additional approach outlined in [SAAR20] provides an easy means to generate new isomorphic involutory or non-involutory MDS matrices.

Till now, we have discussed nonrecursive constructions by direct methods as well as search methods. From the next section onward we discuss recursive constructions by direct methods. We close this section by providing Table 3.2, which summarize the involutory and orthogonal properties of circulant, circulant-like, left-circulant, Toeplitz and Hankel matrices.

Table 3.2: Several results of Circulant, Circulant-like, left-circulant, Toeplitz and Hankel matrices over a finite field ("**DNE**" stands for does not exist).

Type	**Dimension**	**Involutory MDS**	**Orthogonal MDS**
Circulant	$2^n \times 2^n$	DNE	DNE
	$2n \times 2n$	DNE	may exist
	$(2n+1) \times (2n+1)$	DNE	may exist
Type-I	$2n \times 2n$	DNE	DNE
	$(2n+1) \times (2n+1)$	DNE	DNE
Type-II	$2(2n) \times 2(2n)$	DNE	DNE
	$2(2n+1) \times 2(2n+1)$	may exist	may exist
left-Circulant	$2^n \times 2^n$	DNE	DNE
	$2n \times 2n$	may exist	may exist
	$(2n+1) \times (2n+1)$	may exist	may exist
Toeplitz	$2^n \times 2^n$	DNE	DNE
	$2n \times 2n$	DNE	may exist
	$(2n+1) \times (2n+1)$	DNE	may exist
Hankel	$2^n \times 2^n$	DNE	DNE
	$2n \times 2n$	may exist	may exist
	$(2n+1) \times (2n+1)$	may exist	may exist

Remark 3.37. *There was an error in [GR15, Table 1] where it was given that Type-II circulant-like orthogonal MDS matrix of order* $2(2n) \times 2(2n)$ *may exist. We have corrected here in Table 3.2.*

3.7 Recursive MDS Matrices

Before discussing the constructions of recursive MDS matrices, let us recall some definitions and notations that will be used in this section, which mainly focuses on coding theoretic techniques.

Given a polynomial $g(x) = a_0 + a_1x + \ldots + a_{k-1}x^{k-1} + a_kx^k \in \mathbb{F}_q[x]$, where $a_k \neq 0$, the degree of $g(x)$ is denoted as $\deg(g)$ and is equal to k. We say that $g(x)$ is monic if its leading coefficient a_k is equal to 1. The weight of a polynomial corresponds to the number of its nonzero coefficients. The order of a polynomial $g(x)$, where $g(0) \neq 0$, denoted as $\mathrm{ord}(g)$, is the smallest positive integer n for which $g(x)$ divides $x^n - 1$.

Definition 3.2. *Let γ be an element in an extension of $\mathbb{F}_q$. The minimal polynomial of γ over $\mathbb{F}_q$, denoted as $\mathrm{Min}_{\mathbb{F}_q}(\gamma)$, is defined as the monic polynomial $\mu(x) \in \mathbb{F}_q[x]$ with the lowest degree such that $\mu(\gamma) = 0$.*

Let Γ be an $[n, \ell]$ linear code of length n and dimension ℓ over $\mathbb{F}_q$. A generator matrix G of the $[n, \ell]$ code Γ, which has dimensions $\ell \times n$, is said to be in systematic form if it contains the $\ell \times \ell$ identity matrix I_ℓ in its leftmost positions. The remaining part of G, is a matrix of dimension $\ell \times (n - \ell)$. In this section, for convenience, we deviate from the conventional notation and place the identity matrix on the right side in our discussion. Now, we present several useful structural results regarding cyclic codes and MDS codes.

Cyclic codes: An $[n, \ell]$ code is considered to be cyclic code if every cyclic shift of any codeword remains within the code. From an algebraic perspective, cyclic codes can be viewed as ideals of the ring $\mathbb{F}_q[x]/(x^n - 1)$. In other words, each cyclic code Γ of length n can be represented as $\Gamma = \langle g(x) \rangle$, where $g(x)$ is a monic polynomial in $\mathbb{F}_q[x]$ that divides $x^n - 1$. Moreover, $g(x)$ is the unique monic polynomial with the minimum degree within Γ and is referred to as the generator polynomial of Γ. The codewords in Γ are multiples of $g(x)$ with degree less than n, which can be expressed as polynomials $f(x) \in \mathbb{F}_q[x]/(x^n - 1)$ satisfying the condition that $g(x)$ divides $f(x)$. The dimension of the code Γ is determined by $\ell = n - \deg(g)$.

To construct a generator matrix for the code Γ, we can use the following matrix representation:

$$G_1 = \begin{bmatrix} g(x) \\ xg(x) \\ \vdots \\ \underbrace{x^{n-\deg(g)-1}g(x)}_{\text{size } n} \end{bmatrix}.$$

Here, the polynomials $x^i g(x)$ are treated as vectors of length n formed by their coefficients in increasing order of exponents. By applying the division algorithm, we have $x^i = q(x)g(x) + (x^i \mod g(x))$. Therefore, $x^i - (x^i \mod g(x))$ is divisible by $g(x)$, making it a codeword in Γ. Let

$$G = \left[\begin{array}{cc|c} -x^{\deg(g)} & \mod g(x) & 1\ 0\ 0 \ldots 0 \\ -x^{\deg(g)+1} & \mod g(x) & 0\ 1\ 0 \ldots 0 \\ & \vdots & \ddots \\ \underbrace{-x^{n-1}}_{\text{size } \deg(g)} & \underbrace{\mod g(x)}_{\text{size } \deg(g)} & \underbrace{0\ 0\ 0 \ldots 1}_{\text{size } n-\deg(g)} \end{array}\right]. \tag{3.8}$$

It can be observed that matrix G serves as a generator matrix for code Γ since its rows correspond to linearly independent codewords of code Γ.

Remark 3.38. *If* $\gcd(q, n) = 1$*, then* $x^n - 1$ *and its derivative* nx^{n-1} *are relatively prime and thus has no repeated roots in* $x^n - 1$*. Therefore, any polynomial* $g(x)$ *which divides* $x^n - 1$ *must have distinct roots if* $\gcd(q, n) = 1$*.*

Now, we state the following result, which will be beneficial in the subsequent subsections.

Lemma 3.25. *[LN97, Theorem 9.42] Theorem: Consider a monic polynomial* $g(x) \in \mathbb{F}_q[x]$ *of degree* k *with* $\text{ord}(g) = n \geq 2$*. Assume that* g *has distinct roots, say* $\lambda_1, \ldots, \lambda_k \in \bar{\mathbb{F}}_q$*. Then, a polynomial* $f(x) = \sum_{i=0}^{n-1} f_i x^i \in \mathbb{F}_q[x]/(x^n - 1)$ *is a codeword of the cyclic code* $\Gamma = \langle g(x) \rangle$ *if and only if its coefficient vector* $(f_0, f_1, \ldots, f_{n-1})$ *belongs to the null space of the matrix*

$$H = \begin{bmatrix} 1 & \lambda_1 & \lambda_1^2 & \ldots & \lambda_1^{n-1} \\ \vdots & \vdots & \ddots & \vdots & \\ 1 & \lambda_k & \lambda_k^2 & \ldots & \lambda_k^{n-1} \end{bmatrix}.$$

Thus, we can say that H *serves as a parity check matrix for the cyclic code* Γ*.*

Proof. If $f(x)$ is the codeword, $g(x)$ divides $f(x)$. Therefore, $f(\lambda_i) = 0$, that is $f_0 + f_1\lambda_i + f_2\lambda_i^2 + \ldots + f_{n-1}\lambda_i^{n-1} = 0$ for $1 \leq i \leq k$. Thus, $H \cdot [f_0 \ f_1 \ \ldots \ f_{n-1}]^T = 0$.

Conversely, let $f(x) = q(x)g(x) + r(x)$ where $\deg(r) < \deg(g) = k$. Since $f(\lambda_i) = g(\lambda_i) = 0$, therefore $r(\lambda_i) = 0$ for $1 \leq i \leq k$. As $\deg(r) < k$, it cannot have k roots. Thus, $r(x) = 0$ and $g(x)$ divides $f(x)$ which implies $f(x)$ is a codeword. □

3.7.1 Characterization of polynomials that yield recursive MDS matrices

A monic polynomial $g(x) \in \mathbb{F}_q[x]$ of degree k is said to yield a recursive MDS matrix if C_g^m is an MDS matrix for some integer $m \geq k$. This concept is particularly relevant when the size of the diffusion matrix (MDS) is greater than 1. Therefore, we focus on considering polynomials $g(x) \in \mathbb{F}_q[x]$ of degree $k = \deg(g) \geq 2$ when seeking to obtain recursive MDS matrices.

Note that the companion matrix C_g can be interpreted as

$$C_g = \begin{bmatrix} x \\ x^2 \\ \vdots \\ x^{k-1} \\ \underbrace{x^k \bmod g(x)}_{\text{size } k} \end{bmatrix}.$$

Now one can see that

$$C_g^2 = \begin{bmatrix} x^2 \\ x^3 \\ \vdots \\ x^k \bmod g(x) \\ \underbrace{x^{k+1} \bmod g(x)}_{\text{size } k} \end{bmatrix}, \ldots, \quad C_g^m = \begin{bmatrix} x^m \bmod g(x) \\ x^{m+1} \bmod g(x) \\ \vdots \\ x^{m+k-2} \bmod g(x) \\ \underbrace{x^{m+k-1} \bmod g(x)}_{\text{size } k} \end{bmatrix}.$$

According to Remark 2.3, the matrix C_g^m is considered MDS if and only if any set of k columns of the matrix $\bar{G} = [C_g^m \mid I]$ is linearly independent over $\mathbb{F}_q$. Alternatively, the matrix C_g^m is MDS if and only if any set of k columns of the matrix $G' = [-C_g^m \mid I]$ is linearly independent over $\mathbb{F}_q$. The matrix G' can be interpreted in the following way.

$$G' = \left[\begin{array}{cc|c} -x^m & \bmod g(x) & 1\ 0\ 0\ \dots\ 0 \\ -x^{m+1} & \bmod g(x) & 0\ 1\ 0\ \dots\ 0 \\ & \vdots & \ddots \\ \underbrace{-x^{m+k-1} \quad \bmod g(x)}_{\text{size } \deg(g)} & & \underbrace{0\ 0\ 0\ \dots\ 1}_{\text{size } k} \end{array}\right].$$

We now prove the folklore result.

$$C_g^m = \begin{bmatrix} x^m \bmod g(x) \\ x^{m+1} \bmod g(x) \\ \vdots \\ x^{m+k-2} \bmod g(x) \\ x^{m+k-1} \bmod g(x) \end{bmatrix}.$$

We prove it by induction. For $m = 1$,

$$C_g = \begin{bmatrix} x \\ x^2 \\ \vdots \\ x^{k-1} \\ x^k \bmod g(x) \end{bmatrix} = \begin{bmatrix} x \bmod g(x) \\ x^2 \bmod g(x) \\ \vdots \\ x^{k-1} \bmod g(x) \\ x^k \bmod g(x) \end{bmatrix}.$$

Assume it is true for $m = l \geq 1$. Now, we show that it is true for $m = l + 1$.

$$C_g^{l+1} = C_g C_g^l = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & & \ddots & & \vdots \\ 0 & 0 & \dots & \dots & 1 \\ -a_0 & -a_1 & \dots & \dots & -a_{k-1} \end{bmatrix} \begin{bmatrix} x^l \bmod g(x) \\ x^{l+1} \bmod g(x) \\ \vdots \\ x^{l+k-2} \bmod g(x) \\ x^{l+k-1} \bmod g(x) \end{bmatrix}$$

$$= \begin{bmatrix} x^{l+1} \bmod g(x) \\ x^{l+2} \bmod g(x) \\ \vdots \\ x^{l+k-1} \bmod g(x) \\ \sum_{i=0}^{k-1} -a_0 x^{l+i} \bmod g(x) \end{bmatrix}$$

Now, we show that $\sum_{i=0}^{k-1} -a_0 x^{l+i} \bmod g(x) = x^{l+k} \bmod g(x)$.

$$\sum_{i=0}^{k-1} -a_i x^{l+i} \bmod g(x) = \left(x^l \left(\sum_{i=0}^{k-1} -a_i x^i \right) \right) \bmod g(x)$$
$$= (x^l x^k) \bmod g(x) = x^{l+k} \bmod g(x).$$

C_g^k is an MDS matrix if $g(x)$ yields MDS code and $\text{ord}(g) = 2k$. Assume $g(x) \in \mathbb{F}_{2^s}[x]$ has no repeated roots (Subsection 3.7.4 deals when $g(x)$ has repeated roots). If $g(x)$ divides $x^{2k} - 1$, then $g(x)$ divides $x^k - 1$ and hence $\text{ord}(g) \neq 2k$, a contradiction. Therefore, in a field of characteristic 2, if the polynomial $g(x)$ has no repeated roots, then it cannot divide $x^{2k} - 1$ for $\deg(g) = k \geq 2$ which means the length of the code cannot be equal to $2k$. But $g(x)$ will divide $x^{2k+z} - 1$ for some $z > 0$. In such cases, if $g(x)$ yields $[2k + z, k + z, d]$ MDS code, then the distance d will be $2k + z - (k + z) + 1 = k + 1$. To obtain a $[2k, k, k + 1]$ code, the code can be shortened at z positions in such a manner that the recursive structure does not get disturbed. The same idea was proposed by Augot et al. [AF15]. They constructed recursive MDS matrices using the shortened BCH code which we are going to discuss in the next subsection.

3.7.2 Construction of recursive MDS matrices using shortened BCH codes

Definition 3.3. *[GPV17a, Definition 6][MS77, Pages 29, 194, 592] For an $[n, \ell, d]$ code Γ and a set R consisting of z indices $\{i_1, \ldots, i_z\}$, the shortened code Γ_R is defined as the subset of words from Γ that are zero at positions $i_1, \ldots, i_z$, and where the zero coordinates are removed. In other words, the shortened code Γ_R is obtained by effectively shortening the original codewords by z positions.*

As a result, the length of Γ_R is reduced to $n - z$ and the dimension of Γ_R is at least $\ell - z$. Also, the minimal distance of Γ_R is at least d. Observe that the dimension can be greater than $\ell - z$. Consider the code $\Gamma = \{0000, 1011, 0101, 1110\}$. If code is shortened at $z = 2$ positions, 1 and 3 (index starts from 0), the shortened code is $\Gamma_R = \{00, 11\}$. Note that the length, dimension and the distance of Γ_R is $n - z = 4 - 2 = 2$, $1 \geq \ell - z = 2 - 2 = 0$ and $2 \geq d = 2$ respectively.

Consider a monic polynomial $g(x) \in \mathbb{F}_q[x]$ of degree k with $\text{ord}(g) = n \geq 2k$. Let $\mathcal{S}$ be a code with generator matrix $G' = [-C_g^m \mid I]$. It is worth noting that C_g^m is an MDS matrix if and only if $-C_g^m$ is an MDS matrix. By examining (3.8),

we can observe that G' can be viewed as a submatrix of the generator matrix G of the cyclic code $\Gamma = \langle g(x) \rangle$. Specifically, G' is obtained by selecting the k rows with indices $\{m-k, m-k+1, \ldots, m-1\}$ and the $2k$ columns with indices $\{0, 1, \ldots, k-1, m, m+1, \ldots, m+k-1\}$ from G (indices start from 0). Therefore, the code $\mathcal{S}$ can be considered as a shortened code of the cyclic code Γ, obtained by omitting the $n-2k$ positions $\{k, k+1, \ldots, m-1, m+k, m+k+1, \ldots, n-1\}$ from Γ while using G as the generator matrix. It should be noted that $\mathcal{S}$ may not necessarily be a cyclic code.

In the work by Augot et al. [AF15], a technique is proposed for constructing recursive MDS matrices through the proper shortening of appropriate BCH codes. BCH codes belong to a specific class of cyclic codes that offer a guaranteed minimum distance (as defined below). For a more comprehensive understanding of this technique, we recommend referring to Section 3 of [AF15]. Additionally, for further related concepts and background in coding theory, refer to [MS77].

Definition 3.4. *[GPV17a, Definition 7][MS77, Page 202] A BCH code over $\mathbb{F}_q$ is constructed by utilizing an element β from an extension field of $\mathbb{F}_q$. Let $\mathrm{ord}(\beta) = n$ be the order of β. To define a BCH code, we start by selecting integers ℓ and d and consider the $(d-1)$ consecutive powers of β: $\beta^{\ell}, \beta^{\ell+1}, \ldots, \beta^{\ell+d-2}$. Next, we compute the polynomial*

$$g(x) = \mathrm{lcm}\left(\mathrm{Min}_{\mathbb{F}_q}(\beta^{\ell}), \ldots, \mathrm{Min}_{\mathbb{F}_q}(\beta^{\ell+d-2})\right),$$

where $\mathrm{Min}\,\mathbb{F}_q(\gamma)$ represents the minimal polynomial of γ over $\mathbb{F}_q$. The cyclic code, defined by $g(x)$ over $\mathbb{F}_q$, is known as a BCH code. This code has a length of n, a dimension of $n - \deg(g)$, and a minimum distance of at least d.

It is worth emphasizing that when all the conjugates of the elements in the set $\{\beta^{\ell+i} : i = 0, 1, \ldots, d-2\}$ are included within the set itself, the degree of the polynomial $g(x)$ is precisely $d-1$. Consequently, $g(x)$ does not possess any additional roots beyond $\beta^{\ell}, \ldots, \beta^{\ell+d-2}$. In such scenarios, the BCH code defined by $g(x)$ transforms into an MDS code. We can formally state this outcome as a lemma, as follows.

Lemma 3.26. *Suppose we have a BCH code Γ over $\mathbb{F}_q$ defined by k roots $[\beta^{\ell}, \ldots, \beta^{\ell+k-1}]$ and has actual distance of $k+1$, then the code Γ is MDS if and only if the polynomial $P(x) = \prod_{j=0}^{k-1}(x - \beta^{\ell+j})$ belongs to $\mathbb{F}_q[x]$. In this scenario, the generator polynomial $g(x) = \mathrm{lcm}\left(\mathrm{Min}\,\mathbb{F}_q(\beta^{\ell}), \ldots, \mathrm{Min}\,\mathbb{F}_q(\beta^{\ell+k-1})\right)$ is equal to $P(x)$.*

Proof. If part: Let $P(x) = \prod_{j=0}^{k-1}(x - \beta^{\ell+j})$. If $P(x) \in \mathbb{F}_q[x]$, then $P(x)$ contains all conjugates of its roots and so $g(x) = P(x)$. The degree of $g(x)$ is k, the dimension of

the code is $n-k$ and the actual distance is $\geq k+1$. But from the Singleton bound, the actual distance is $\leq n-(n-k)+1 = k+1$. Therefore, the actual distance is $k+1$ and achieves the Singleton bound. Hence, it is MDS.

Only if part: Let

$$g(x) = \text{lcm}\left(\text{Min}_{\mathbb{F}_q}(\beta^{\ell}), \ldots, \text{Min}_{\mathbb{F}_q}(\beta^{\ell+k-1})\right)$$

generates the MDS BCH code. The dimension of the code is $n - \deg(g)$ and the actual distance is $k+1$ which is equal to $n-(n-\deg(g))+1$ (from the Singleton bound). Therefore, $k = \deg(g)$ which implies the set of k roots $[\beta^{\ell}, \ldots, \beta^{\ell+k-1}]$ contains all its conjugates. Thus, $P(x) = \prod_{j=0}^{k-1}(x-\beta^{\ell+j}) \in \mathbb{F}_q[x]$. □

In Lemma 3.26, the BCH code Γ must have the condition that the actual distance is $k+1$, otherwise it may not satisfy the sufficient condition. We show it by one example. Consider $q = 2^3$. Take $\beta \in \mathbb{F}_{q^2}$ such that $\text{ord}(\beta) = 9$. Let

$$g(x) = \text{lcm}\left(\text{Min}_{\mathbb{F}_q}(\beta^3), \text{Min}_{\mathbb{F}_q}(\beta^4), \text{Min}_{\mathbb{F}_q}(\beta^5)\right).$$

The cyclotomic coset mod n over $\mathbb{F}_q$ is $C_0 = \{0\}, C_1 = \{1,8\}, C_2 = \{2,7\}, C_3 = \{3,6\}, C_4 = \{4,5\}$. Therefore,

$$g(x) = \text{Min}_{\mathbb{F}_q}(\beta^3) \cdot \text{Min}_{\mathbb{F}_q}(\beta^4).$$

Let

$$g'(x) = \text{lcm}\left(\text{Min}_{\mathbb{F}_q}(\beta^3), \text{Min}_{\mathbb{F}_q}(\beta^4), \text{Min}_{\mathbb{F}_q}(\beta^5), \text{Min}_{\mathbb{F}_q}(\beta^6)\right).$$

The degree of $g(x)$ is 4 which yields $[n=9, \ell=5, d\geq 5]_8$ code. The distance is ≥ 5 because $g(x)$ and $g'(x)$ yield the same code and the designed distance of $g'(x)$ is 5. By the Singleton bound, $d \leq n-l+1 = 9-5+1 = 5$. Thus, $g(x)$ yields $[9,5,5]_8$ code which is MDS. But $P(x) = \prod_{j=3}^{5}(x-\beta^j)$ is not in $\mathbb{F}_q[x]$ because the actual distance is 5, not $k+1=4$.

The example above demonstrates the necessity of the actual distance $k+1$ in Lemma 3.26. The lemma mentioned in ([AF15, Lemma 1]) does not make assumption on the distance of the BCH code and thus suffers a gap in the statement. We investigated that lemma again and provide the correct statement in Lemma 3.26.

It is important to emphasize that in a BCH code, the roots of its generating polynomial may not always be consecutive powers of an element. For example, consider the

BCH code generated by the polynomial $g(x) \in \mathbb{F}_q[x]$ where $q = 2^3$ defined by

$$g(x) = \text{lcm}\left(\text{Min}_{\mathbb{F}_q}(\beta^2), \text{Min}_{\mathbb{F}_q}(\beta^3)\right).$$

Roots of $g(x)$ are $\beta^2, \beta^3, \beta^6, \beta^7$ which are not consecutive in powers of β.

In [AF15, GPV17a], authors considered a particular kind of BCH codes, called c-BCH codes, where all the roots of its generating polynomial are consecutive powers of some element in some field. It is worth to point out that the authors in [AF15] also used c-BCH code without mentioning it explicitly.

Definition 3.5. *[GPV17a, Definition 8] A c-BCH code over $\mathbb{F}_q$ is defined as a BCH code whose generating polynomial has roots that can be expressed as consecutive powers of an element β in an extension field of $\mathbb{F}_q$. Based on Lemma 3.26, it is evident that a c-BCH code over $\mathbb{F}_q$ is an MDS code. Therefore, we use the term MDS c-BCH codes to refer to such codes.*

It is worth noting that the MDS c-BCH code over $\mathbb{F}_q$ defined by $g(x)$ has a length of $n = \text{ord}(g)$ and a dimension of $k = n - \deg(g)$. Therefore, the corresponding MDS matrix would have a size of $k \times \deg(g)$. However, using such an MDS matrix as a diffusion layer may not be ideal unless $\deg(g) = k$, as the input and output sizes of a diffusion layer are typically the same, resulting in $n = 2k$. Also, it is not possible to have $\text{ord}(\beta) = 2k$ since elements of even order do not exist in extensions of $\mathbb{F}_2$. To address this issue, Augot et al.[AF15] suggest using a shortened MDS c-BCH code (see Definition 3.3) instead of a full length MDS c-BCH code with $n > 2k$. Based on the earlier discussion, we can observe that the generating polynomial of an MDS c-BCH code yields a recursive MDS matrix. In the following discussion, we will explore the technique proposed in [AF15] for obtaining recursive MDS matrices of size k by appropriately shortening MDS c-BCH codes.

The approach involves searching for $[n = 2k + z, m = k + z, d = k + 1]$ MDS c-BCH codes and then shortening them on z positions to obtain the desired $[2k, k, k+1]$ MDS codes, where z is an odd integer. The first step is to construct a c-BCH code of length $n = 2k + z$ (which is limited by $q + 1$ according to the assumption that the MDS conjecture holds, as stated in Fact 2.1). A suitable β of order n in an extension field of $\mathbb{F}_q$ is selected, along with ℓ where $0 \leq \ell < n$. Then, the polynomial $P(x) = \prod_{j=0}^{k-1}(x - \beta^{\ell+j})$ is computed. The lemma mentioned earlier provides a condition under which the polynomial $P(x)$ generates an MDS c-BCH code. The next step is to verify whether this condition is satisfied. If it is, we can obtain a recursive MDS matrix from the generating polynomial. Therefore, the crucial aspect is to check the condition:

$$P(x) = \prod_{j=0}^{k-1}(x - \beta^{\ell+j}) \in \mathbb{F}_q[x]. \tag{3.9}$$

If this condition holds true, it implies that our choice of n, β, and ℓ yields an MDS c-BCH code, and its generating polynomial is precisely equal to $P(x)$.

In subsequent observations (as described in the paragraph following Theorem 3.15), it is noted that if there exists a choice of n along with β and ℓ, such that an MDS c-BCH code (with length n and dimension $n-k$) is obtained, then for any β^i where $\gcd(i, n) = 1$, we can also obtain an MDS c-BCH code. This implies that if there exists an ℓ satisfying Equation 3.9 for a particular choice of n, then we consider n as a successful choice. Similarly, we call the pair (n, ℓ) a successful choice. To enumerate all MDS c-BCH codes over $\mathbb{F}_q$ that can be constructed using this method, Augot et al. developed an algorithm (refer to [AF15, Section 4.2]) that verifies the condition by computing the polynomial $P(x)$ for all candidate values within the specified ranges given by $n = 2k + z \leq (q + 1)$ with z being odd, β having order n, and $0 \leq \ell < n - 1$.

Efficient Approach for Identifying All MDS c-BCH Codes

The main limitation of the algorithm proposed by Augot et al. is the potential occurrence of unsuccessful choices for the parameters n and ℓ within the specified ranges. In some cases, it is not possible to find an MDS c-BCH code (of length $n = 2k + z$ and dimension $n - k = k + z$ over $\mathbb{F}_q$) for any ℓ in the range $0 \leq \ell < n - 1$. Consequently, computing and verifying the polynomial $P(x)$ for such choices becomes unnecessary. Additionally, for unsuccessful choices of n, the computation must be performed in extension fields of $\mathbb{F}_q$, which can be computationally intensive. Additionally, the algorithm may compute the same polynomials twice for certain successful choices of n.

In [GPV17a], Gupta et al. discussed the values of n and the corresponding values of ℓ that yield generating polynomials for MDS c-BCH codes. The most significant result was Theorem 3.13 which gives a nice relation between n and q. By getting so, a lot of unnecessary choices of n could be omitted and finally we get a set of only those possible values of n which would definitely yield MDS c-BCH codes. This theorem was significant not only because it would directly give the possible values of n, but at the same time the value of l could also be determined. To obtain all possible values of n and l was a remarkable improvement as it drastically reduces the running time of

finding all MDS c-BCH of length n over $\mathbb{F}_q$ which was not possible from the algorithm proposed by Augot et al. in [AF15].

We make the assumption that $k \geq 2$ and $n = 2k + z$ ($\leq q + 1$) for a odd integer z (see Fact 2.1).

Theorem 3.13. *[GPV17a, Theorem 2] For integers k and n satisfying $k \geq 2$ and $n > 2k$, an MDS c-BCH code with length n and dimension $(n - k)$ over $\mathbb{F}_q$ exists if and only if $q \equiv \pm 1$ modulo n.*

In [AF15], the algorithm would search for all candidate values of n by choosing z from 1 to $q + 1 - 2k$. As Theorem 3.13 suggests, many of these values of z were definitely wrong choices and hence these values would increase the running time of the algorithm. Moreover, one could obtain the exact values of l depending upon whether $n \mid q - 1$ or $n \mid q + 1$ from the proof of Theorem 3.13. When n divides $q - 1$, l can take any value between 0 and $n - 1$. On the other hand, when n divides $q + 1$, the value of l depends on whether k is even or odd. If k is even, then l is equal to $(n - k + 1)/2$. If k is odd, then l is equal to $n - (k - 1)/2$. These conditions lead to a formula for determining the number of such MDS c-BCH codes.

Theorem 3.14. *[GPV17a, Theorem 3] When n divides $(q - 1)$, the count of MDS c-BCH codes with a length of n and a dimension of $(n-k)$ over $\mathbb{F}_q$ is given by $n \cdot \frac{\phi(n)}{2}$, where $\phi(n)$ represents Euler's totient function.*

Theorem 3.15. *[GPV17a, Theorem 4] When n divides $(q + 1)$, the count of MDS c-BCH codes with a length of n and a dimension of $(n - k)$ over $\mathbb{F}_q$ is given by $\dfrac{\phi(n)}{2}$.*

For proofs of Theorems 3.14 and 3.15, see [GPV17a]. Nevertheless, we present a brief discussion on the number of MDS c-BCH codes obtained in these theorems. In Theorem 3.14, the number of MDS c-BCH codes is $n \cdot \dfrac{\phi(n)}{2}$. The term n appears due to the choices of l which varies from 0 to $n - 1$, i.e. n choices, whereas the term $\phi(n)/2$ appears because of the number of choices of β (see Equation 3.9) whose order must be exactly n. There are $\phi(n)$ choices of such β. But β and β^{-1} yield the same code, that's why the number of choices of β becomes $\phi(n)/2$. In Theorem 3.15, there is only one choice of l (see the paragraph after Theorem 3.13) and $\phi(n)/2$ choices of β (similar argument as for Theorem 3.14).

3.7.3 Recursive MDS matrices using the parity check matrix

Lemma 3.27. *[GPV17b, Lemma 4] Consider a monic polynomial $g(x) \in \mathbb{F}_q[x]$ of degree k with $\mathrm{ord}(g) = n$. Let the polynomial g have k distinct roots de-*

noted as $\lambda_1, \ldots, \lambda_k \in \bar{\mathbb{F}}_q$. *Now, let* $\mathcal{S}$ *be the code with generator matrix* $G' = [-C_g^m \mid I]$ *for some value of* m *where* $k \leq m \leq n-k$. *In this case, a vector* $(f_0, f_1, \ldots, f_{k-1}, f_m, f_{m+1}, \ldots, f_{m+k-1}) \in \mathbb{F}_q^{2k}$ *belongs to the code* $\mathcal{S}$ *if and only if it belongs to the null space of the matrix* H' *defined by*

$$H' = \begin{bmatrix} 1 & \lambda_1 & \ldots & \lambda_1^{k-1} & \lambda_1^m & \lambda_1^{m+1} & \ldots & \lambda_1^{m+k-1} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \lambda_k & \ldots & \lambda_k^{k-1} & \lambda_k^m & \lambda_k^{m+1} & \ldots & \lambda_k^{m+k-1} \end{bmatrix}. \tag{3.10}$$

Proof. The code generated by the matrix G' is the shortened code $\mathcal{S}$ of the code $\Gamma = \langle g(x) \rangle$ shortened at positions $R = \{k, k+1, \ldots, m-1, m+k, m+k+1, \ldots, n-1\}$. Therefore, $(f_0, f_1, \ldots, f_{k-1}, f_m, f_{m+1}, \ldots, f_{m+k-1})$ is an element of $\mathcal{S}$ if and only if $(f_0, f_1, f_2, \ldots, f_{n-1})$ is in Γ where $(f_k, f_{k+1}, \ldots, f_{m-1}, f_{m+k}, f_{m+k+1}, \ldots, f_{n-1}) = (0, 0, \ldots, 0, 0, 0, \ldots, 0)$. From Lemma 3.25, $(f_0, f_1, f_2, \ldots, f_{n-1})$ is a codeword of $\Gamma = \langle g(x) \rangle$ if and only if $\sum_{j=0}^{n-1} f_j \lambda_i^j = 0$ for $1 \leq i \leq k$. Suppose $(f_0, f_1, f_2, \ldots, f_{n-1})$ is shortened at positions R. Let $\mathcal{I} = \{1, 2, \ldots, n\}$. Then $\sum_{j \in \mathcal{I} \setminus R} f_j \lambda_i^j = 0$ if and only if $\sum_{j=0}^{n-1} f_j \lambda_i^j = 0$ for $1 \leq i \leq k$. Hence, $(f_0, f_1, \ldots, f_{k-1}, f_m, f_{m+1}, \ldots, f_{m+k-1})$ is in the null space of the matrix H' if and only if $(f_0, f_1, f_2, \ldots, f_{n-1})$ is in Γ or $(f_0, f_1, \ldots, f_{k-1}, f_m, f_{m+1}, \ldots, f_{m+k-1})$ is in $\mathcal{S}$. □

Consider the code $\mathcal{S}$ and the matrix H' as defined in the previous lemma. It is important to note that the code $\mathcal{S}$ has a dimension of k. The code $\mathcal{S}$ is classified as an MDS code, with a minimum distance of $k+1$, if and only if the null space of the matrix H' in $\mathbb{F}_q^{2k}$ does not contain a nonzero vector of weight k or less. This means that every set of k columns from the matrix H' must be linearly independent over $\mathbb{F}_q$ [MS77]. To summarize this result, we state the following theorem.

Theorem 3.16. *[GPV17b, Theorem 2] Consider a monic polynomial* $g(x) \in \mathbb{F}_q[x]$ *of degree* k *with* $\text{ord}(g) = n$. *Let* $\lambda_1, \ldots, \lambda_k \in \bar{\mathbb{F}}_q$ *be the* k *distinct roots of* g. *Let* m *be an integer satisfying* $k \leq m \leq n-k$. *The matrix* $M = C_g^m$ *is an MDS matrix if and only if any set of* k *columns of the matrix* H', *as defined in Equation 3.10, is linearly independent over* $\mathbb{F}_q$.

Theorem 3.16 can be proved alternatively as shown in [GPV15]. Suppose $g(x)$ has k distinct roots $\lambda_1, \ldots, \lambda_k$. The idea is to use the fact that $C_g = VDV^{-1}$ or $C_g^T = (V^T)^{-1} D V^T$ where $V = vand[\lambda_1, \lambda_2, \ldots, \lambda_k]$ and $D = diag[\lambda_1, \lambda_2, \ldots, \lambda_k]$. If C_g^m is MDS, then $(C_g^m)^T = (C_g^T)^m$ is MDS and thus any k columns of $[I \mid (C_g^T)^m]$ are linearly

independent. Now,

$$[I \mid (C_g^T)^m] = [I \mid (V^T)^{-1} D^m V^T] = (V^T)^{-1}[V^T \mid D^m V^T] = (V^T)^{-1} H'$$

where $H' = [V^T \mid D^m V^T]$. As a result, $(C_g^T)^m$ is MDS and so C_g^m if and only if any set of k columns of H' is linearly independent.

Lemma 3.28. *[GPV17b, Corollary 1] If the polynomial $g(x) = x^k + a_{k-1}x^{k-1} + \ldots + a_1 x + a_0 \in \mathbb{F}_q[x]$ (where $a_0 \neq 0$) yields a recursive MDS matrix, then its (monic) reciprocal polynomial*

$$g^*(x) = \frac{x^k}{a_0} g\left(\frac{1}{x}\right) = x^k + \frac{a_1}{a_0}x^{k-1} + \ldots + \frac{a_{k-1}}{a_0}x + \frac{1}{a_0}$$

also yields a recursive MDS matrix.

Proof. The matrix $C_{g^*} = R(C_g)^{-1}R$ where

$$R = \begin{bmatrix} 0 & 0 & \ldots & 0 & 1 \\ 0 & 0 & \ldots & 1 & 0 \\ & & \ldots & & \\ 0 & 1 & \ldots & 0 & 0 \\ 1 & 0 & \ldots & 0 & 0 \end{bmatrix}$$

and $R^2 = I_k$. The matrix $C_{g^*}^m = R(C_g^{-1})^m R$ is MDS if and only if $(C_g^{-1})^m$ is MDS and it is true because $(C_g^{-1})^m$ is MDS if and only if C_g^m is MDS. □

Observe that if λ is the root of g, then λ^{-1} is the root of g^* provided $\lambda \neq 0$.

Lemma 3.29. *[GPV17b, Corollary 2] Let us consider a polynomial $g(x) = \prod_{i=1}^{k}(x - \lambda_i) \in \mathbb{F}_q[x]$ that yields a recursive MDS matrix. Then, for any nonzero element $c \in \mathbb{F}_q$, the polynomial $c^k g\left(\frac{x}{c}\right) = \prod_{i=1}^{k}(x - c\lambda_i)$ also yields a recursive MDS matrix.*

Proof. Let $g^*(x) = c^k g\left(\frac{x}{c}\right)$. The matrix $C_{g^*} = cDC_gD^{-1}$ where

$$D = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 & 0 \\ 0 & c & 0 & \dots & 0 & 0 \\ 0 & 0 & c^2 & \dots & 0 & 0 \\ & & & \dots & & \\ 0 & 0 & 0 & \dots & c^{k-2} & 0 \\ 0 & 0 & 0 & \dots & 0 & c^{k-1} \end{bmatrix}$$

The matrix $C_{g^*}^m = c^m DC_g^m D^{-1}$ is MDS if and only if C_g^m is MDS. □

By applying the two previous lemmas, we can generate additional polynomials that yield recursive MDS matrices from an initial polynomial. However, it is worth exploring other techniques that can provide us with more polynomials yielding recursive MDS matrices based on the initial polynomial.

Now, we will discuss five methods for constructing polynomials that yield recursive MDS matrices. These methods ensure that the constructed polynomials have distinct roots. The key tool in these methods is Theorem 3.16, where we carefully choose values of λ_i for $1 \leq i \leq k$ and verify that the polynomial $g(x) = \prod_{i=1}^{k}(x - \lambda_i) \in \mathbb{F}_q[x]$ satisfies the condition stated in Theorem 3.16. To demonstrate this, we need to show that any k-column submatrix of H' obtained from (3.10) is nonsingular. In other words, we aim to prove that the determinant of the matrix

$$H'[R] = \begin{bmatrix} \lambda_1^{r_1} & \lambda_1^{r_2} & \dots & \lambda_1^{r_k} \\ \lambda_2^{r_1} & \lambda_2^{r_2} & \dots & \lambda_2^{r_k} \\ \vdots & \vdots & \ddots & \vdots \\ \lambda_k^{r_1} & \lambda_k^{r_2} & \dots & \lambda_k^{r_k} \end{bmatrix}, \tag{3.11}$$

is nonzero for any subset $R = \{r_1, r_2, \dots, r_k\} \subset E = \{0, 1, \dots, k-1, m, m+1, \dots, m+k-1\}$ of cardinality k.

Construction I(a)

In this method, we exploit the use of consecutive powers of an element, or a fixed multiple of them, to construct a polynomial that yields a recursive MDS matrix. It is interesting to note that this approach shares similarities with BCH codes. However, this method allows us to obtain a larger set of recursive MDS matrices compared to the approach utilizing shortened BCH codes proposed by Augot et al. [AF15].

Theorem 3.17. *[GPV17b, Theorem 3] Consider the polynomial $g(x) = \prod_{i=1}^{k}(x-\lambda_i)$, where $\lambda_i = \theta^{i-1}$ for $1 \leq i \leq k$ for some $\theta \in \mathbb{F}_q^*$. For an integer $m \geq k$, the matrix C_g^m is MDS if and only if $\theta^i \neq \theta^j$ for all $i, j \in E$, where $E = \{0, 1, \ldots, k-1, m, m+1, \ldots, m+k-1\}$ with $i \neq j$.*

Proof. As mentioned earlier, the matrix C_g^m is an MDS matrix if and only if the determinant of $H'[R]$ is nonzero for all subsets $R = \{r_1, r_2, \ldots, r_k\} \subset E$. We have $\lambda_i = \theta^{i-1}$ for $1 \leq i \leq k$, and so we get

$$H'[R] = \begin{bmatrix} 1 & 1 & \ldots & 1 \\ \theta^{r_1} & \theta^{r_2} & \ldots & \theta^{r_k} \\ \vdots & \vdots & \ddots & \vdots \\ (\theta^{k-1})^{r_1} & (\theta^{k-1})^{r_2} & \ldots & (\theta^{k-1})^{r_k} \end{bmatrix} = \begin{bmatrix} 1 & 1 & \ldots & 1 \\ \theta^{r_1} & \theta^{r_2} & \ldots & \theta^{r_k} \\ \vdots & \vdots & \ddots & \vdots \\ (\theta^{r_1})^{k-1} & (\theta^{r_2})^{k-1} & \ldots & (\theta^{r_k})^{k-1} \end{bmatrix}.$$

Let $y_{r_i} = \theta^{r_i}$ for $1 \leq i \leq k$. Therefore, we have $\det(H'[R]) = \det(V)$, where

$$V = \begin{bmatrix} 1 & 1 & \ldots & 1 \\ y_{r_1} & y_{r_2} & \ldots & y_{r_k} \\ \vdots & \vdots & \ddots & \vdots \\ y_{r_1}^{k-1} & y_{r_2}^{k-1} & \ldots & y_{r_k}^{k-1} \end{bmatrix}.$$

Here, V is a Vandermonde matrix, and $\det(V) \neq 0$ if and only if the values $y_{r_i} = \theta^{r_i}$, where $1 \leq i \leq k$, are distinct and nonzero. Therefore, the matrix C_g^m is MDS if and only if the values θ^{r_i}, where $1 \leq i \leq k$, are distinct for all $R = \{r_1, r_2, \ldots, r_k\} \subset E$. This condition is equivalent to $\theta^i \neq \theta^j$ for all $i, j \in E = \{0, 1, \ldots, k-1, m, m+1, \ldots, m+k-1\}$ with $i \neq j$. Thus, the theorem holds. □

Note that if θ satisfies the condition in Theorem 3.17, it is necessary for the order of θ to be greater than or equal to $2k$. Moreover, this condition is also sufficient when $m = k$.

Example 3.9. *Let α be a primitive element of $\mathbb{F}_{2^8}$ with $\alpha^8+\alpha^4+\alpha^3+\alpha^2+1 = 0$. If we consider $\theta = \alpha$ in Theorem 3.17, then we have $\lambda_1 = 1$, $\lambda_2 = \alpha$, $\lambda_3 = \alpha^2$, and $\lambda_4 = \alpha^3$. We get $g_1(x) = (x+1)(x+\alpha)(x+\alpha^2)(x+\alpha^3) = x^4 + \alpha^{75}x^3 + \alpha^{249}x^2 + \alpha^{78}x + \alpha^6$. Then the companion matrix of g_1 is $C_{g_1} = \mathsf{Companion}(\alpha^6, \alpha^{78}, \alpha^{249}, \alpha^{75})$ and*

$$C_{g_1}^4 = \begin{bmatrix} \alpha^6 & \alpha^{78} & \alpha^{249} & \alpha^{75} \\ \alpha^{81} & \alpha^{59} & \alpha^{189} & \alpha^{163} \\ \alpha^{169} & \alpha^{162} & \alpha^{198} & \alpha^{131} \\ \alpha^{137} & \alpha^{253} & \alpha^{49} & \alpha^{143} \end{bmatrix}$$

is an MDS matrix.

We can obtain many more polynomials using Theorem 3.17 and Lemma 3.29. In the previous example, if we multiply $c = \alpha$ *with* λ_i *for* $1 \leq i \leq 4$*, we get* $\lambda_1 = \alpha, \lambda_2 = \alpha^2, \lambda_3 = \alpha^3$ *and* $\lambda_4 = \alpha^4$*. Then we get* $g_1(x) = (x+\alpha)(x+\alpha^2)(x+\alpha^3)(x+\alpha^4) = x^4 + \alpha^{76}x^3 + \alpha^{251}x^2 + \alpha^{81}x + \alpha^{10}$*. Then the companion matrix of* g_1 *is* $C_{g_1} = \mathsf{Companion}(\alpha^{10}, \alpha^{81}, \alpha^{251}, \alpha^{76})$ *and*

$$C_{g_1}^4 = \begin{bmatrix} \alpha^{10} & \alpha^{81} & \alpha^{251} & \alpha^{76} \\ \alpha^{86} & \alpha^{63} & \alpha^{192} & \alpha^{165} \\ \alpha^{175} & \alpha^{167} & \alpha^{202} & \alpha^{134} \\ \alpha^{144} & \alpha^4 & \alpha^{54} & \alpha^{147} \end{bmatrix}$$

which is again an MDS matrix.

In the design of the PHOTON family of hash functions, the following matrices (see [GPP11, Table 1]) over $\mathbb{F}_{2^4}$ can be derived from the construction described above. The field $\mathbb{F}_{2^4}$ is constructed using the polynomial $x^4 + x + 1$ and α is a root of this polynomial.

1. The polynomial $f_5(x) = x^5 + \alpha x^4 + (\alpha^3+1)x^3 + (\alpha^3+1)x^2 + \alpha x + 1$ yields a recursive MDS matrix of order 5 over $\mathbb{F}_{2^4}$. It can be observed that the roots of f_5 are the consecutive powers of β: $\{\beta^{13}, \beta^{14}, \beta^0, \beta^1, \beta^2\}$, where $\beta = \alpha^4$. The polynomial $f_5(x)$ can be obtained by choosing $\theta = \beta$ and $c = \beta^{13}$ in Theorem 3.17.

2. The polynomial $f_6(x) = x^6 + \alpha x^5 + \alpha^3 x^4 + (\alpha^2+1)x^3 + \alpha^3 x^2 + \alpha x + 1$ yields a recursive MDS matrix of order 6 over $\mathbb{F}_{2^4}$. It can be checked that the roots of f_6 are the consecutive powers of $\gamma : \{\gamma^6, \gamma^7, \gamma^8, \gamma^9, \gamma^{10}, \gamma^{11}\}$, where $\gamma = (\beta+1)^{45}$, β is a root of the irreducible polynomial $x^8 + x^4 + x^3 + x + 1$, $\alpha = \beta^7 + \beta^6 + \beta^5 + 1$, and the order of γ is equal to 17. By selecting $\theta = \gamma$ and $c = \gamma^6$ in Theorem 3.17, the polynomial $f_6(x)$ can be obtained.

3. The polynomial $f_7(x) = x^7 + \alpha^2 x^6 + (\alpha^2+\alpha)x^5 + x^4 + x^3 + (\alpha^2+\alpha)x^2 + \alpha^2 x + 1$ yields a recursive MDS matrix of order 7 over $\mathbb{F}_{2^4}$. It can be observed that the roots

of f_7 correspond to the consecutive powers of α : $\{\alpha^{12}, \alpha^{13}, \alpha^{14}, \alpha^0, \alpha^1, \alpha^2, \alpha^3\}$. By selecting $\theta = \alpha$ and $c = \alpha^{12}$ in Theorem 3.17, the polynomial $f_7(x)$ can be obtained.

4. The polynomial $f_8(x) = x^8 + (\alpha^2 + \alpha)x^7 + (\alpha^2 + 1)x^6 + \alpha^3 x^5 + \alpha x^4 + (\alpha^3 + \alpha + 1)x^3 + \alpha x^2 + \alpha^2 x + \alpha$ yields a recursive MDS matrix of order 8 over $\mathbb{F}_{2^4}$. It can be verified that the roots λ_i's of f_8 belong to $\mathbb{F}_{2^8}$ and follow the pattern $\lambda_i = \theta^{i-1}c, 1 \leq i \leq 8$, where $\theta = (\beta + 1)^{15}$ and $c = (\beta + 1)^{109}$ are determined based on β being a root of the irreducible polynomial $x^8 + x^4 + x^3 + x + 1$ and $\alpha = \beta^7 + \beta^6 + \beta^5 + 1$. The order of θ is equal to 17. By selecting these specific values of λ_i in Theorem 3.17, the polynomial $f_8(x)$ can be obtained.

Relationship with [AF15]: Augot et al. introduced a method for constructing recursive MDS matrices using shortened BCH codes. The method involves computing a polynomial $g(x)$ by selecting a suitable element β in an extension field of $\mathbb{F}_q$. The roots of $g(x)$ are consecutive powers of β, denoted as $\beta^i, \beta^{i+1}, \ldots, \beta^{i+k-1}$ for some integer i. When the roots of $g(x)$ are conjugate to each other, the polynomial $g(x)$ belongs to $\mathbb{F}_q[x]$, ensuring that the corresponding BCH code with generator polynomial $g(x)$ is MDS. Consequently, $g(x)$ yields a recursive MDS matrix. By selecting $\theta = \beta$ and $c = \beta^i$ in Theorem 3.17, we can further establish that $g(x)$ generates a recursive MDS matrix.

Remark 3.39. *It is important to note that if $\lambda_1 = c$ is not a power of θ in Theorem 3.17, then the resulting polynomial may not be a generator polynomial of a BCH code. An example of such a polynomial is $f_8(x)$ mentioned in Item 4 above. In that case, the roots of $f_8(x)$ are given by $\lambda_i = \alpha^2\theta^{4+i}$ for $1 \leq i \leq 8$. It can be verified that these roots cannot be expressed as consecutive powers of an element in $\mathbb{F}_{2^8}$. Consequently, this method yields a larger set of polynomials that generate recursive MDS matrices compared to the method using shortened BCH codes. These additional polynomials can be considered as those obtained by applying Lemma 3.29.*

In the following, we present two additional methods for constructing polynomials that yield recursive MDS matrices. These constructions bear some resemblance to the first method, but they are distinct and will be denoted as I(b) and I(c), respectively.

Construction I(b)

Theorem 3.18. *[GPV19, Theorem 3] Consider the polynomial $g(x) = \prod_{i=1}^{k}(x - \lambda_i)$, where $\lambda_i = \theta^{i-1}$ for $1 \leq i \leq k-1$ and $\lambda_k = \theta^k$ for some $\theta \in \mathbb{F}_q^*$. For an integer*

$m \geq k$, the matrix C_g^m is MDS if and only if $\theta^r \neq \theta^{r'}$ for $r, r' \in E$ and $\sum_{i=1}^{k} \theta^{r_i} \neq 0$ for all $R = \{r_1, r_2, \ldots, r_k\} \subset E$, where $E = \{0, 1, \ldots, k-1, m, m+1, \ldots, m+k-1\}$.

Proof. The matrix C_g^m is MDS if and only if the determinant of the matrix $H'[R]$ (as defined in (3.11)) is nonzero for all subsets $R = \{r_1, r_2, \ldots, r_k\}$ of the set E. We have $\lambda_i = \theta^{i-1}$ for $1 \leq i \leq k-1$ and $\lambda_k = \theta^k$. So we have $H'[R] =$

$$\begin{bmatrix} 1 & 1 & \ldots & 1 \\ \theta^{r_1} & \theta^{r_2} & \ldots & \theta^{r_k} \\ \vdots & \vdots & \ddots & \vdots \\ (\theta^{k-2})^{r_1} & (\theta^{k-2})^{r_2} & \ldots & (\theta^{k-2})^{r_k} \\ (\theta^k)^{r_1} & (\theta^k)^{r_2} & \ldots & (\theta^k)^{r_k} \end{bmatrix} = \begin{bmatrix} 1 & 1 & \ldots & 1 \\ \theta^{r_1} & \theta^{r_2} & \ldots & \theta^{r_k} \\ \vdots & \vdots & \ddots & \vdots \\ (\theta^{r_1})^{k-2} & (\theta^{r_2})^{k-2} & \ldots & (\theta^{r_k})^{k-2} \\ (\theta^{r_1})^k & (\theta^{r_2})^k & \ldots & (\theta^{r_k})^k \end{bmatrix}.$$

Let $y_{r_i} = \theta^{r_i}$ for $1 \leq i \leq k$. Thus, we can express the determinant of $H'[R]$ as the determinant of V', which is given by:

$$V' = \begin{bmatrix} 1 & 1 & \ldots & 1 \\ y_{r_1} & y_{r_2} & \ldots & y_{r_k} \\ \vdots & \vdots & \ddots & \vdots \\ y_{r_1}^{k-2} & y_{r_2}^{k-2} & \ldots & y_{r_k}^{k-2} \\ y_{r_1}^{k} & y_{r_2}^{k} & \ldots & y_{r_k}^{k} \end{bmatrix}$$

Now observe that the matrix V' is a generalized Vandermonde matrix of the form $V_\perp(\mathbf{y}; I)$ with $I = \{1\}$. By applying Corollary 2.10, we can conclude that $\det(V') \neq 0$ if and only if the values y_{r_i} are distinct and $\sum_{i=1}^{k} y_{r_i} \neq 0$. This completes the proof. □

Remark 3.40. *It can be observed that the condition on θ in Theorem 3.18 remains applicable even if we choose $\lambda_i = \theta^{i-1}c$ for $1 \leq i \leq k-1$ and $\lambda_k = \theta^k c$ for some $c \in \mathbb{F}_q^*$. By considering the roots in this manner, the resulting polynomials are equivalent to those obtained by applying Lemma 3.29.*

Construction I(c)

Corollary 3.9. *[GPV19, Corollary 1] Consider the polynomial $g(x) = \prod_{i=1}^{k}(x - \lambda_i)$, where $\lambda_1 = 1$, and $\lambda_i = \theta^i$, $2 \leq i \leq k$ for some $\theta \in \mathbb{F}_q^*$. For an integer $m \geq k$, the matrix C_g^m is MDS if and only if $\theta^r \neq \theta^{r'}$ for $r, r' \in E$ and $\sum_{i=1}^{k} \theta^{-r_i} \neq 0$ for all $R = \{r_1, r_2, \ldots, r_k\} \subset E$, where $E = \{0, 1, \ldots, k-1, m, m+1, \ldots, m+k-1\}$.*

Proof. Consider $\gamma_i = \lambda_{k-i+1} = (\theta^{-1})^{i-1}c$ for $1 \leq i \leq k-1$ and $\gamma_k = \lambda_1 = (\theta^{-1})^k c$, where $c = \theta^k$. By Theorem 3.18 and the previous remark, the matrix C_g^m is MDS if and only if θ^{-r_i}, for $1 \leq i \leq k$, are distinct, and $\sum_{i=1}^{k} \theta^{-r_i} \neq 0$ for all $R = \{r_1, r_2, \ldots, r_k\} \subset E$. This completes the proof. □

Remark 3.41. *The proof of the above corollary can also be seen similarly to the proof of Theorem 3.18 derived using Corollary 2.11.*

Remark 3.42. *As a consequence of the previous results, we can derive an infinite class of polynomials that yield recursive MDS matrices. Consider $s \geq 2k$ and α as a root of an irreducible polynomial of degree s over $\mathbb{F}_2$. For any $c \in \bar{\mathbb{F}}_{2^s}^*$, the polynomial $g(x) = (x - c\alpha^k) \cdot \prod_{i=0}^{k-2}(x - c\alpha^i)$ yields recursive MDS matrices C_g^m for $m \in \{k, \ldots, s-k\}$.*

In the following examples, the constructing polynomial for $\mathbb{F}_{2^8}$ is given by $x^8 + x^7 + x^6 + x + 1$, and α is one of its roots.

Example 3.10. *We observe that $\beta = \alpha^{15}$ is a primitive 17th root of unity, and the degree of its minimal polynomial is 8. Based on the previous remark, the polynomial $g(x) = (x-1)(x-\beta)(x-\beta^2)(x-\beta^4)$ yields a recursive MDS matrix of order 4. It can be observed that C_g^m is MDS for $4 \leq m \leq 13$. Interestingly, this polynomial can also be obtained using construction II(b) for recursive MDS matrices.*

Example 3.11. *Consider the polynomial $g(x) = (x-1)(x-\beta)(x-\beta^2)(x-\beta^3)(x-\beta^5)$, where $\beta = \alpha^{15}$ is a primitive 17th root of unity. It has been verified that this polynomial satisfies the condition in Theorem 3.18, and thus it yields a recursive MDS matrix of order 5. It can be checked that C_g^5 is an MDS matrix. Furthermore, it can also be verified that this polynomial cannot be obtained using the other known constructions such as I(a), II(a), and II(b).*

Next, we present two additional methods, denoted as II(a) and II(b), for the construction of polynomials that yield recursive MDS matrices. These constructions share similarities but are not identical.

Construction II(a)

The recursive MDS matrices obtained using the method that utilizes Gabidulin codes (described in Section 3.4 of [Ber13]) can also be obtained using the following alternative method. Consider a subfield $\mathbb{F}_{q_1}$ of $\mathbb{F}_q$, where $q_1 = p^{s'}$ and s' divides s.

Theorem 3.19. *[GPV17b, Theorem 4] Consider the polynomial* $g(x) = \prod_{i=1}^{k}(x - \lambda_i)$, *where* $\lambda_i = \theta^{q_1^{i-1}}$ *for* $1 \leq i \leq k$, *and* $\theta \in \mathbb{F}_q^*$. *Let* $E = \{0, 1, \ldots, k-1, m, m+1, \ldots, m+k-1\}$ *for some integer* $m \geq k$. *The matrix* C_g^m *is MDS if and only if any subset* B *of* $\theta^i : i \in E$ *with* $|B| = k$ *is linearly independent over* $\mathbb{F}_{q_1}$.

Remark 3.43. *We can observe that the condition on* θ *in Theorem 3.19 is applicable even if we choose* $\lambda_i = \theta^{q_1^{i-1}} c$ *for some* $c \in \mathbb{F}_q^*$. *By considering the roots in this manner, we obtain the same set of polynomials as those obtained by applying Lemma 3.29. Additionally, it can be seen, as mentioned in Remark 3.39, that it is not always possible to represent* $\lambda_i = \theta^{q_1^{i-1}} c, 1 \leq i \leq k$, *in the form* $\theta'^{q_1^{j-1}}, 1 \leq j \leq k$, *for some* θ'. *Therefore, this method allows us to obtain a larger set of polynomials compared to the method that utilizes Gabidulin codes (as described in [Ber13, Section 3.4]).*

Example 3.12. *Let the field* $\mathbb{F}_{2^8} = \mathbb{F}_2[x]/\langle \mu(x) \rangle$, *where* $\mu(x) = x^8 + x^4 + x^3 + x^2 + 1$ *is a primitive polynomial over* $\mathbb{F}_2$. *Let* $\alpha \in \mathbb{F}_{2^8}$ *be a root of* $\mu(x)$, *i.e. a primitive element of* $\mathbb{F}_{2^8}$. *Let us consider* $q_1 = 2, m = k = 4$ *and* $\theta = \alpha$ *in Theorem 3.19, then* $\lambda_1 = \alpha, \lambda_2 = \alpha^2, \lambda_3 = \alpha^4$ *and* $\lambda_4 = \alpha^8$. *Then we get* $g_2(x) = (x + \alpha)(x + \alpha^2)(x + \alpha^4)(x + \alpha^8) = x^4 + \alpha^{238}x^3 + \alpha^{235}x^2 + \alpha^{168}x + \alpha^{15}$. *Thus, the companion matrix* $C_{g_2} = \mathsf{Companion}(\alpha^{15}, \alpha^{168}, \alpha^{235}, \alpha^{238})$ *and*

$$C_{g_2}^4 = \begin{bmatrix} \alpha^{15} & \alpha^{168} & \alpha^{235} & \alpha^{238} \\ \alpha^{253} & \alpha^{49} & \alpha^{170} & \alpha^{190} \\ \alpha^{205} & \alpha^{246} & \alpha^{92} & \alpha^{138} \\ \alpha^{153} & \alpha^{252} & \alpha^{3} & \alpha^{18} \end{bmatrix}$$

is an MDS matrix.

Relationship with [Ber13]: It has been observed in [Ber13] that MDS matrices can be constructed using Gabidulin codes [BO04] by appropriately selecting certain parameters. Furthermore, it was established that such an MDS matrix can be transformed into a recursive MDS matrix by choosing a polynomial basis. We will now demonstrate that Berger's method is a specific case of Theorem 3.19. Let $q = 2^s$ with $s = 2k$. The generator matrix, as described in [Ber13, Section 3.4], can be represented

as

$$G = [V \mid \hat{V}] = \left[\begin{array}{cccc|cccc} 1 & \alpha & \dots & \alpha^{k-1} & \alpha^{k} & \alpha^{k+1} & \dots & \alpha^{2k-1} \\ 1 & \alpha^{2} & \dots & \alpha^{2(k-1)} & \alpha^{2k} & \alpha^{2(k+1)} & \dots & \alpha^{2(2k-1)} \\ 1 & \alpha^{4} & \dots & \alpha^{4(k-1)} & \alpha^{4k} & \alpha^{4(k+1)} & \dots & \alpha^{4(2k-1)} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \alpha^{2^{k-1}} & \dots & \alpha^{2^{k-1}(k-1)} & \alpha^{2^{k-1}k} & \alpha^{2^{k-1}(k+1)} & \dots & \alpha^{2^{k-1}(2k-1)} \end{array}\right],$$

where $\{1, \alpha, \alpha^2, \dots, \alpha^{2k-1}\}$ is a polynomial basis of $\mathbb{F}_q$. Subsequently, the matrix G can be transformed into systematic form $[I \mid A]$ by applying elementary row operations: $V^{-1}G = V^{-1}[V \mid \hat{V}] = [I \mid A]$. It has been demonstrated that the first column of matrix $A = V^{-1}\hat{V}$ provides the coefficients of the polynomial g. Notably, $C_g^k = A$ represents an MDS matrix.

It is worth noting that when $m \geq k$ and $\deg(\mathrm{Min}_{\mathbb{F}_{q_1}}(\theta)) \geq m+k$, where $\mathrm{Min}_{\mathbb{F}_{q_1}}(\theta)$ denotes the minimal polynomial of θ over $\mathbb{F}_{q_1}$, the elements θ^{r_i} for $1 \leq i \leq k$ are linearly independent over $\mathbb{F}_{q_1}$ for any subset $R = \{r_1, r_2, \dots, r_k\} \subset E = \{0, 1, 2, \dots, k-1, m, m+1, \dots, m+k-1\}$. By choosing θ as a primitive element in $\mathbb{F}_{2^s}$ with $2k \leq s$ and setting $m = k$ and $q_1 = 2$, we satisfy the condition of Theorem 3.19. The polynomials obtained using these choices in Theorem 3.19 are equivalent to those obtained by the method discussed in [Ber13, Section 3.4]. Therefore, we can generate an infinite class of polynomials that yield recursive MDS matrices. Furthermore, the codes associated with matrices of this type possess the additional property of maximum rank distance. As mentioned in Remark 3.43, there are many other choices that can be accommodated in Theorem 3.19, resulting in a larger set of polynomials compared to the method discussed in [Ber13, Section 3.4]. It is important to note that the condition $\deg(\mathrm{Min}_{\mathbb{F}_{q_1}}(\theta)) \geq m + k$ is not necessary. We will now provide an example where an element θ satisfies the condition in Theorem 3.19, but $\deg(\mathrm{Min}_{\mathbb{F}_{q_1}}(\theta)) < m + k$.

Example 3.13. *Let the field $\mathbb{F}_{2^6} = \mathbb{F}_2[x]/\langle\mu(x)\rangle$, where $\mu(x) = x^6 + x^4 + x^3 + x + 1$ is a primitive polynomial over $\mathbb{F}_2$. Let $\alpha \in \mathbb{F}_{2^6}$ be a root of $\mu(x)$, i.e. a primitive element of $\mathbb{F}_{2^6}$. Let us consider $q_1 = 2, m = k = 4, \theta = \alpha$ in Theorem 3.19, then $\lambda_1 = \alpha, \lambda_2 = \alpha^2, \lambda_3 = \alpha^4$ and $\lambda_4 = \alpha^8$. Then we get $g_3(x) = (x + \alpha)(x + \alpha^2)(x + \alpha^4)(x + \alpha^8) = x^4 + \alpha^{30}x^3 + \alpha^{43}x^2 + \alpha^{51}x + \alpha^{15}$. Thus, the companion matrix $C_{g_3} =$*

$\mathsf{Companion}(\alpha^{15}, \alpha^{51}, \alpha^{43}, \alpha^{30})$ *and*

$$C_{g_3}^4 = \begin{bmatrix} \alpha^{15} & \alpha^{51} & \alpha^{43} & \alpha^{30} \\ \alpha^{45} & \alpha^{28} & \alpha^{34} & \alpha^{19} \\ \alpha^{34} & \alpha^{40} & \alpha^{43} & \alpha^{5} \\ \alpha^{20} & \alpha^{17} & \alpha^{47} & \alpha^{42} \end{bmatrix}$$

is an MDS matrix. It can be observed that $\deg(\mathrm{Min}_{\mathbb{F}_2}(\alpha)) = 6 < m + k = 8$.

Construction II(b)

We now introduce an alternative approach that bears resemblance to the previous method. However, the key distinction lies in the fact that the roots of the polynomial $g(x)$ discussed below may not conform to the format described in Theorem 3.19 (also refer to Remark 3.44). By employing this technique, we can generate a fresh set of infinite polynomials that result in recursive Maximum Distance Separable (MDS) matrices. Here, we consider $\mathbb{F}_{q_1}$ as a subfield of $\mathbb{F}_q$, which implies that $q_1 = p^{s'}$ for some s' dividing s.

Theorem 3.20. *[GPV17b, Theorem 5] Given* $\lambda_1 = 1$ *and* $\lambda_i = \theta^{q_1^{i-2}}$, $2 \leq i \leq k$, *consider the polynomial* $g(x) = \prod_{i=1}^{k}(x - \lambda_i)$, *where* θ *is an element in* $\mathbb{F}_q^*$. *Let* $E = \{0, 1, \ldots, k-1, m, m+1, \ldots, m+k-1\}$, *where* m *is an integer such that* $m \geq k$. *Then the matrix* C_g^m *is MDS if and only if for any subset* $B = \{b_1, b_2, \ldots, b_k\}$ *of* $\{\theta^i : i \in E\}$, *where* $|B| = k$, *there is no nontrivial linear combination over* $\mathbb{F}_{q_1}$ *that satisfies* $\sum_{j=1}^{k} \alpha_j b_j = 0 \quad$ *and* $\quad \sum_{j=1}^{k} \alpha_j = 0$.

Remark 3.44. *It is worth noting that the condition on* θ *in Theorem 3.20 remains valid even when we set* $\lambda_1 = c$ *and* $\lambda_i = \theta^{q_1^{i-2}} c$ *for* $2 \leq i \leq k$, *where* $c \in \mathbb{F}_q^*$. *By considering the roots in this manner, the resulting polynomials are identical to those obtained by applying Lemma 3.29.*

Example 3.14. *Let the field* $\mathbb{F}_{2^8} = \mathbb{F}_2[x]/\langle \mu(x) \rangle$, *where* $\mu(x) = x^8 + x^4 + x^3 + x^2 + 1$ *is a primitive polynomial over* $\mathbb{F}_2$. *Let* $\alpha \in \mathbb{F}_{2^8}$ *be a root of* $\mu(x)$, *i.e. a primitive element of* $\mathbb{F}_{2^8}$. *Let us consider* $q_1 = 2, m = k = 4, \theta = \alpha$ *in Theorem 3.20, then* $\lambda_1 = 1, \lambda_2 = \alpha, \lambda_3 = \alpha^2$ *and* $\lambda_4 = \alpha^4$. *Then we get* $g_4(x) = (x+1)(x+\alpha)(x+\alpha^2)(x+\alpha^4) = x^4 + \alpha^{129}x^3 + \alpha^{167}x^2 + \alpha^{11}x + \alpha^7$. *Thus, the companion matrix* $C_{g_4} =$

$\mathsf{Companion}(\alpha^7, \alpha^{11}, \alpha^{167}, \alpha^{129})$ *and*

$$C_{g_4}^4 = \begin{bmatrix} \alpha^7 & \alpha^{11} & \alpha^{167} & \alpha^{129} \\ \alpha^{136} & \alpha^2 & \alpha^{77} & \alpha^{121} \\ \alpha^{128} & \alpha^{232} & \alpha^{47} & \alpha^{54} \\ \alpha^{61} & \alpha^{120} & \alpha^{211} & \alpha^{81} \end{bmatrix}$$

is an MDS matrix.

Remark 3.45. *It has been verified that the roots* $\lambda_1 = 1, \lambda_2 = \alpha, \lambda_3 = \alpha^2$ *and* $\lambda_4 = \alpha^4$ *of* $g_4(x)$ *mentioned in the previous example cannot be represented as* $\theta'^{q_1^{i-1}} c'$ *for any* $\theta', c' \in \mathbb{F}_{2^8}$ *(see Remark 3.43). As a result, it is not possible to derive the polynomial* $g_4(x)$ *using Theorem 3.19 and Lemma 3.29.*

It is worth noting that if an element $\theta \in \bar{\mathbb{F}}_q^*$ satisfies the condition specified in Theorem 3.19, then it also satisfies the condition outlined in Theorem 3.20. Consequently, if we have a polynomial $g(x) = \prod_{i=1}^{k}(x - c\theta^{q_1^{i-1}})$ obtained through the Theorem 3.19 and Remark 3.43, we can observe that the polynomial $g'(x) = (x-c)\prod_{i=2}^{k}(x - c\theta^{q_1^{i-2}})$ also yields a recursive MDS matrix. This can be achieved by employing Theorem 3.20 and Remark 3.44. As a result, using the choices presented in [Ber13, Section 3.4], we obtain a new infinite class of recursive MDS matrices through the utilization of Theorem 3.20.

3.7.4 Repeated-root cyclic codes

We were interested to find $g(x)$ which would yield MDS code and $\mathrm{ord}(g) = 2k$ (recall Subsection 3.7.1) so that C_g^k becomes MDS matrix. For the case when $g(x)$ has no repeated root in the field of characteristic 2, the idea was to generate a $[2k+z, k+z, d]$ MDS code and then shorten the code at z positions to get a $[2k, k, d]$ code. The reason for shortening was due to the fact that $g(x)$ cannot have order $2k$ in the field of characteristic 2. But $g(x)$ may have the order $2k$ when it has some repeated roots. We show it by one example.

Suppose $q = 2$ and $k = 7$. Consider $g(x) = (x^3 + x^2 + 1)^2(x + 1)$. It is easy to check that $g(x)$ divides $x^{14} + 1$ because $x^7 + 1 = (x+1)(x^3+x^2+1)(x^3+x+1)$. Now we show that $g(x)$ does not divide $x^i + 1$ for $1 \leq i \leq 13$. For $1 \leq i \leq 6$, it is obvious because $\deg(g) = 7$. Moreover $g(x)$ does not divide $x^7 + 1$ (look at the factors). Now, if $g(x)$ divides $x^i + 1$ for $8 \leq i \leq 13$, then $g(x)$ divides $x^{14-i} + 1$, a contradiction. Thus, $\mathrm{ord}(g) = 14 = 2k$. Note that $g(x)$ has repeated roots, otherwise it would not be possible.

This subsection discusses the case when $g(x)$ has multiple roots. One of the most important results of this subsection is Corollary 3.10 which states that if $g(x) \in \mathbb{F}_2[x]$ has any repeated root and $\deg(g) \geq 2$, it is not possible to get a recursive MDS matrix $M = C_g^m$ for any $m \geq 0$. As a result, it can be concluded that there does not exist any involutory recursive MDS matrix of order $k \geq 2$ over the field of characteristic 2. Now we define some basic notions and discuss some existing results which are going to be used in this subsection. We present only those results on repeated-root cyclic codes which are relevant to this chapter. For more details, please see [CMSvS91, VL91].

Definition 3.6. *[GPV17b, Definition 1] Consider a polynomial $g(x) = \sum_{i=0}^{k} a_i x^i \in \mathbb{F}_q[x]$. For some non-negative integer d, we define the dth* Hasse derivative *$g^{[d]}(x)$ of g as follows:*

$$g^{[d]}(x) = \sum_{i=0}^{k} \binom{i}{d} a_i x^{i-d},$$

where we assume $\binom{i}{d} = 0$ if $i < d$.

The lemma presented below is of significant importance.

Lemma 3.30. *[CMSvS91, Section II] Consider an irreducible polynomial $\mu(x) \in \mathbb{F}_q[x]$ and a positive integer e. Then $\mu(x)^e$ divides a polynomial $g(x) \in \mathbb{F}_q[x]$ if and only if $\mu(x)$ divides $g(x)$ as well as its first $e-1$ Hasse derivatives. In other words, $\mu(x)$ divides $g^{[d]}(x)$ for all d satisfying $0 \leq d \leq e-1$.*

When the generator polynomial $g(x) \in \mathbb{F}_q[x]$ possesses a repeated root (i.e. a root with multiplicity greater than 1), the resulting code $\Gamma = \langle g(x) \rangle$ derived from g is referred to as a repeated-root cyclic code. The comprehensive theory of such codes was introduced in [CMSvS91, VL91]. By referring to Lemma 3.30, one can observe that an equivalent result to the aforementioned lemma, in the case of repeated-root cyclic codes, is stated below (also see [CMSvS91, Section II]).

Lemma 3.31. *[GPV17b, Lemma 3] Consider a monic polynomial $g(x) \in \mathbb{F}_q[x]$ of degree k and $\mathrm{ord}(g) = n \geq 2$. Let g have t distinct roots, denoted as $\lambda_1, \ldots, \lambda_t \in \bar{\mathbb{F}}_q$, with respective multiplicities $e_1, e_2, \ldots, e_t$. Then $f(x) = \sum_{i=0}^{n-1} f_i x^i \in \mathbb{F}_q[x]/(x^n - 1)$ is a codeword of $\Gamma = \langle g(x) \rangle$ if and only if the coefficient vector $(f_0, f_1, \ldots, f_{n-1})$ of f belongs to the null space of the matrix*

$$H = \begin{bmatrix} H_{e_1}(\lambda_1) \\ \vdots \\ H_{e_t}(\lambda_t) \end{bmatrix},$$

where the matrix $H_{e_i}(\lambda_i)$ *is of size* $e_i \times n$ *and its rows are the* n*-tuples*

$$\left(\binom{0}{j}, \binom{1}{j}\lambda_i^{1-j}, \binom{2}{j}\lambda_i^{2-j}, \ldots, \binom{n-1}{j}\lambda_i^{n-j-1}\right) \tag{3.12}$$

for $0 \leq j \leq e_i - 1$*. Or, in other words,* H *is a parity check matrix of* Γ*.*

The n-tuples referenced in the previous lemma correspond to the jth Hasse derivatives of the vector $(1, \lambda_i, \lambda_i^2, \ldots, \lambda_i^{n-1})$ treating λ_i as a variable (see Definition 3.6).

Lemma 3.32. *[GPV17b, Lemma 4] Consider a monic polynomial* $g(x) \in \mathbb{F}_q[x]$ *of degree* k*, where* $\mathrm{ord}(g) = n$*. Let* g *have* t *distinct roots denoted as* $\lambda_1, \ldots, \lambda_t \in \bar{\mathbb{F}}_q$*, with respective multiplicities* $e_1, e_2, \ldots, e_t$*. Consider the code* $\mathcal{S}$ *with a generator matrix* $G' = [-C_g^m \mid I]$*, where* m *is an integer such that* $k \leq m \leq n - k$*. Then a vector* $(f_0, f_1, \ldots, f_{k-1}, f_m, f_{m+1}, \ldots, f_{m+k-1}) \in \mathbb{F}_q^{2k}$ *belongs to the code* $\mathcal{S}$ *if and only if it belongs to the null space of the matrix* H' *given by*

$$H' = \begin{bmatrix} H'_{e_1}(\lambda_1) \\ \vdots \\ H'_{e_t}(\lambda_t) \end{bmatrix}, \tag{3.13}$$

where the matrix $H'_{e_i}(\lambda_i)$ *is of size* $e_i \times 2k$ *and its rows are the* $2k$*-tuples*

$$\left(\binom{0}{j}, \binom{1}{j}\lambda_i^{1-j}, \ldots, \binom{k-1}{j}\lambda_i^{k-1-j}, \binom{m}{j}\lambda_i^{m-j}, \binom{m+1}{j}\lambda_i^{m+1-j}, \ldots, \binom{m+k-1}{j}\lambda_i^{m+k-j-1}\right)$$

for $0 \leq j < e_i$*. Specifically, when the polynomial* $g(x)$ *has no repeated roots (i.e.* $e_i = 1$ *for* $1 \leq i \leq t = k$*), the matrix* H' *can be expressed as follows:*

$$H' = \begin{bmatrix} 1 & \lambda_1 & \ldots & \lambda_1^{k-1} & \lambda_1^m & \lambda_1^{m+1} & \ldots & \lambda_1^{m+k-1} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \lambda_k & \ldots & \lambda_k^{k-1} & \lambda_k^m & \lambda_k^{m+1} & \ldots & \lambda_k^{m+k-1} \end{bmatrix}. \tag{3.14}$$

Consider $\mathcal{S}$ and H' as defined in the aforementioned lemma. It is worth noting that the code $\mathcal{S}$ has dimension of k. Based on Lemma 3.32, it becomes evident that the code $\mathcal{S}$ is MDS, indicating that it has a minimum distance of $k+1$, if and only if the null space of the matrix H' in $\mathbb{F}_q^{2k}$ does not contain any nonzero vector of weight k or lower. In other words, this condition can be equivalently expressed as the linear independence of any k columns of the matrix H' over $\mathbb{F}_q$. For the sake of convenience, we summarize this result as follows.

Theorem 3.21. *[GPV17b, Theorem 2] Consider a monic polynomial* $g(x) \in \mathbb{F}_q[x]$

of degree k and $\mathrm{ord}(g) = n$. Let g have t distinct roots denoted as $\lambda_1, \ldots, \lambda_t \in \bar{\mathbb{F}}_q$, with respective multiplicities $e_1, e_2, \ldots, e_t$. Suppose we have an integer m such that $k \leq m \leq n - k$. The matrix $M = C_g^m$ is an MDS matrix if and only if any set of k columns of the matrix H', as defined in Equation 3.13, is linearly independent over $\mathbb{F}_q$.

Corollary 3.10. *[GPV17b, Corollary 3] Assume that $char(\mathbb{F}_q) = p$, and let $g(x) \in \mathbb{F}_q[x]$ be a monic polynomial of degree k. Suppose $g(0) \neq 0$ and $\mathrm{ord}(g) = n$. If the polynomial $g(x)$ possesses a root $\lambda \in \bar{\mathbb{F}}_q$ with a multiplicity $e \geq p$, then it can be concluded that the matrix $M = C_g^m$ is not MDS for any non-negative integer m.*

Remark 3.46. *A notable consequence of the above corollary is that if $char(\mathbb{F}_q) = 2$, then a polynomial $g(x) \in \mathbb{F}_q[x]$ that possesses a repeated root cannot yields a recursive MDS matrix. Consequently, recursive MDS matrices over fields with characteristic 2, which hold significant importance in cryptographic applications, can only be obtained from polynomials without repeated roots. However, if $char(\mathbb{F}_q) = p$ where p is an odd prime, it is possible to find polynomials $g(x) \in \mathbb{F}_q[x]$ with repeated roots that yield recursive MDS matrices. An illustrative example is provided below.*

Example 3.15. *Suppose $q = p = 7$. Let's consider the polynomial $g(x) = (x-1)^3 = x^3 - 3x^2 - 4x - 1 \in \mathbb{F}_7[x]$. The matrix C_g^3 derived from $C_g = \mathsf{Companion}(1, 4, 3)$ of g is indeed an MDS matrix*

$$C_g = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 4 & 3 \end{bmatrix} \quad \text{and} \quad C_g^3 = \begin{bmatrix} 1 & 4 & 3 \\ 3 & 6 & 6 \\ 6 & 6 & 3 \end{bmatrix}.$$

Involutory Recursive MDS matrices

Recall that a square matrix M is said to be involutory if $M^2 = I$. An involutory recursive MDS matrix M is an MDS matrix which is involutory and equal to C_g^m for some companion matrix C_g and some $m \geq k$.

Theorem 3.22. *[GPV19, Theorem 2] Over fields of characteristic 2, there exists no involutory recursive MDS matrix M, except in the trivial case where $M = [1]$.*

Proof. Let $M = C_g^m$ be a recursive MDS matrix for some polynomial over $\mathbb{F}_q$ of degree k and $m \geq k$. Let l be the smallest positive integer such that $C_g^l = I_k$. Then, from the discussion in the above section if M is MDS then g cannot have a repeated root. In that case $l = \mathrm{ord}(g)$ is odd since $char(\mathbb{F}_q) = 2$ (see the first paragraph of Subsection

3.7.4). If suppose $M^2 = C_g^{2m} = I$ then l must divide m. Therefore, $M = C_g^m = I$ which is a contradiction as I_k cannot be MDS unless $I_k = [1]$. □

3.7.5 Search vs. direct construction method

In the year 2011, in order to reduce the hardware area of the proposed hash function family PHOTON [GPP11], Guo et al. came out with an idea of reducing the diffusion layer area by choosing a sparse non-MDS matrix and then used it recursively to finally obtain an MDS one. The proposed sparse matrix was a companion matrix. The PHOTON family has different sizes of companion matrices ranging from 4×4 upto 8×8. These matrices were obtained through exhaustive search and the search was possible because the order of the matrix was limited to 8 and the field size was limited to 2^8.

Authors of [GPP11] defined $Serial(z_0, \ldots, z_{k-1})$ ($\mathsf{Companion}(z_0, \ldots, z_{k-1})$), which is the companion matrix of $z_0 + z_1x + z_2x^2 + \ldots + z_{k-1}x^{k-1} + x^k$. Their objective was to find suitable candidates so that $\mathsf{Companion}(z_0, \ldots, z_{k-1})^k$ forms an MDS matrix. In [GPP11], authors proposed the MDS matrix $\mathsf{Companion}(1, \alpha, 1, \alpha^2)^4$ over $\mathbb{F}_{2^8}$, where α represents a root of the irreducible polynomial $x^8 + x^4 + x^3 + x + 1$, for AES MixColumn operation, offering a compact design and improved hardware efficiency [GPP11]. It is worth noting that the MixColumn operation described in [GPP11] consists of applying the matrix $\mathsf{Companion}(z_0, \ldots, z_{k-1})$ to the input column vector k times. More formally, let $W = (w_0, \ldots, w_{k-1})^T$ denote the input column vector for MixColumn, and let $Y = (y_0, \ldots, y_{k-1})^T$ be the corresponding output. We can express this operation as

$$Y = A^k \times W = \underbrace{(A \times (A \times \ldots \times (A}_{\text{k times}} \times W))),$$

where $A = \mathsf{Companion}(z_0, \ldots, z_{k-1})$. Consequently, the hardware circuitry will rely on the companion matrix A rather than the MDS matrix A^k. Note that authors of [GPP11] employed MAGMA [BCP97] to exhaustively test all possible values of z_0, z_1, z_2 and z_3 and found $\mathsf{Companion}(1, \alpha, 1, \alpha^2)$ to be the right candidate, which gives an MDS matrix when raised to the power 4. Authors of [SDMS12, WWW13] proposed new $k \times k$ recursive MDS matrices based on companion matrices for smaller values of k.

To conquer the quest of getting recursive MDS matrices of bigger size, a more concrete theory was required and that appeared from the realm of coding theory.

Though it seems an obvious choice, the initial few works following the work of Guo et al. lacked this direction. In the year 2013, Berger [Ber13] showed how to use the Gabidulin codes [BO04] to obtain infinite class of recursive MDS matrices. Then next year, in 2014, Augot et al. [AF15] showed how to use shortened BCH codes to obtain recursive MDS matrices. These two methods were enough to provide recursive MDS matrix of any order, nevertheless a more generic construction criterion appeared in [GPV17b] using the roots of the characteristic polynomial of the companion matrix. In fact, any recursive MDS matrix obtained using the shortened BCH or the Gabidulin code could also be obtained from the generic construction method discussed in [GPV17b]. Now, we have enough construction methods for larger matrix size also.

Though the generic construction methods open the feasibility of obtaining recursive MDS matrices of any order, there is no guarantee of getting a matrix of the optimal hardware area; even for the smaller size, it is not guaranteed. The only known method, so far, which could produce the optimal matrix in terms of area is the exhaustive search which is feasible when the matrix size is small (for example, say upto 8) and the field size is not too large (say upto $\mathbb{F}_{2^8}$). Of course, the exhaustive search is never a solution for matrices of larger order or bigger field size.

Broadly, there are two different ways to do exhaustive search: (a) try with all possible field elements or (b) try with only a subset of the field elements which require less hardware area. The search method (a) is naive and obvious and it works for matrices of only small orders (say upto 8). But this method always produces the optimal matrices. Whereas the second method (b) may work for larger order matrices but its feasibility depends upon the cardinality and choice of the subset of field elements which will be used for searching. Smaller the cardinality, less chance to obtain required matrices, even though they exist in that field. The main reason is that there may not exist any recursive MDS matrix with the entries chosen from the subset, but could exist if constructed from elements other than subset elements.

A lot of work has been done in this direction. We are not going to discuss the search methods for constructing recursive MDS matrices in this chapter, however, they are particularly interesting in the context of lightweight cryptography. In Chapter 5, we will explore a search method for providing recursive MDS matrices from a newly proposed sparse matrix. For additional constructions by search method, we recommend referring to [GPP11, GR13b, SDMS12, TTKS18, WWW13].

3.8 Conclusion

In this chapter, we have studied the properties of MDS matrices and its various constructions from Cauchy, Vandermonde, circulant, left-circulant, Toeplitz, Hankel and companion matrices. We find a nontrivial equivalence between the Cauchy based constructions and its corresponding Vandermonde based constructions. We also observe a interconnection that a left-circulant matrix is nothing but a row-permutated circulant matrix and a similar connection between Hankel and Toeplitz matrices. Using the interconnection we provide an alternative proof that left-circulant and Hankel matrices of order 2^n are not both MDS and involutory. We do not discuss efficiency issues but the theory accumulated and discussed here should provide an idea towards efficiency. We revisit the results discussed in [AF15] and find a gap in one of the lemmas in that paper. In Subsection 3.7.2, we show the existence of gap by providing an example and then provide the proof of the lemma after rephrasing it correctly. In Subsection 3.7.3, we describe five approaches for constructing polynomials that yield recursive MDS matrices. The key components utilized in these approaches are Theorem 3.16 and determinant of a matrix as defined in (3.11). We believe that more constructions are possible using Theorem 3.16 and (3.11) and so it may be taken as a future research.

4

A Study of Recursive MDS Matrix Construction Using Low Fixed XOR Matrices

Contents

4.1 Introduction

The advantage of recursive MDS matrices is their suitability for lightweight implementations, as the diffusion layer can be implemented efficiently by recursively executing the implementation of the sparse matrices, which requires a small number of clock cycles.

To yield a recursive MDS matrix, a matrix must have at least one nonzero element in each row and each column. So, we consider an $n \times n$ matrix M of the form $P(D + A') = PD + A$, where P represents a permutation matrix, D is a nonsingular diagonal matrix and A' contains t ($t \geq 1$) nonzero elements in some non-diagonal position. Thus, fixed XOR (see Section 2.6.1) of M is t. We will call such matrix as t-XOR matrix.

Some known t-XOR matrices are companion, DSI and sparse DSI matrices. For an $n \times n$ companion and DSI matrix, t is $n - 1$ whereas for sparse DSI matrix, t is

$\lceil n/2 \rceil$. Since a sparse DSI matrix has the least t value among companion, DSI and sparse DSI, that is why it is more suited for the lightweight diffusion layer.

In this chapter, we systematically study low fixed XOR matrices. We begin by examining 1-XOR matrices and provide an upper limit on the count of nonzero elements in an $n \times n$ 1-XOR matrix when it is raised to the power of n. We then shift our focus to 2-XOR matrices and present results that demonstrate the non-existence of 2-XOR matrices of order 5 and 6 that are 5-MDS and 6-MDS, respectively. These results are significant because they provide lower bounds on the number of fixed XORs needed for n-MDS lower XOR matrices of order n for $n \leq 6$. Additionally, this chapter introduces some new mathematical results while also rediscovering existing results on DSI and sparse DSI matrices. Finally, we prove that an 8-MDS sparse DSI matrix of order 8 over the field $\mathbb{F}_{2^8}$ does not exist. This result remained unsolved in [TTKS18] due to the extensive search space.

Outline: The rest of this chapter is structured as follows: In Section 4.2, we study t-XOR matrices for constructing recursive MDS matrices with $t = 1$ and $t = 2$. In Section 4.3, we present new mathematical results and rediscover some existing results on DSI and sparse DSI matrices. Additionally, this section demonstrates the non-existence of an 8-MDS sparse DSI matrix of order 8 over the field $\mathbb{F}_{2^8}$. Finally, in Section 4.4, we conclude the chapter and discuss possible directions for future research.

4.2 t-XOR Matrices

Companion matrices, which are t-XOR matrices with $t = n-1$, have been thoroughly studied for constructing recursive MDS matrices using both search methods [GPP11, GPPR11, GR13b, TTKS18] and direct methods [AF15, GPV17b, GPV17a, GPV19, KPSV21]. However, less attention has been given to the study of t-XOR matrices with $t < n - 1$. In this section, we will focus on the study of t-XOR matrices with $t = 1$ and $t = 2$.
In the following lemma, we study an equivalence relation between the t-XOR matrices.

Lemma 4.1. *Let M_1 be a t-XOR matrix of order $n \geq 2$. Then M_1 is permutation equivalent to some t-XOR matrix $M_2 = QD' + A'$, where Q is a permutation matrix, D' is a nonsingular diagonal matrix and A' has t nonzero elements in its first t rows.*

Proof. Let $M_1 = PD + A$ be a t-XOR matrix, where A has $r_1, r_2, \ldots, r_k$ nonzero

elements in the $i_1, i_2, \ldots, i_k$-th row respectively such that $r_1 + r_2 + \cdots + r_k = t$ and $k \leq t$.

Now consider the permutation matrix P_1 obtained from the identity matrix by permuting the row i_1 to row 1, row i_2 to row 2, ..., row i_k to row k. Now

$$\begin{aligned} P_1 M_1 P_1^{-1} &= P_1(PD + A)P_1^{-1} \\ &= P_1 P D P_1^{-1} + P_1 A P_1^{-1}. \end{aligned}$$

Since $DP_1^{-1} = P_1^{-1}D'$ for some diagonal matrix D', we have

$$\begin{aligned} P_1 M_1 P_1^{-1} &= P_1 P P_1^{-1} D' + P_1 A P_1^{-1} \\ &= QD' + A', \end{aligned}$$

where $Q = P_1 P P_1^{-1}$ and $A' = P_1 A P_1^{-1}$. Also note that A' has altogether t nonzero elements in its 1st, 2nd, ..., k-th row. Let $M_2 = QD' + A'$. Therefore, M_1 is permutation equivalent to M_2. □

Example 4.1. *For example, consider a* 1*-XOR matrix of order* 4

$$M_1 = \begin{bmatrix} b & 0 & 0 & 0 \\ 0 & c & d & 0 \\ 0 & 0 & e & 0 \\ 0 & 0 & 0 & f \end{bmatrix} \text{ and } P_1 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Then $P_1 M_1 P_1^{-1} = M_2$*, where* M_2 *is a* 1*-XOR matrix given by*

$$M_2 = \begin{bmatrix} c & 0 & d & 0 \\ 0 & b & 0 & 0 \\ 0 & 0 & e & 0 \\ 0 & 0 & 0 & f \end{bmatrix}.$$

Remark 4.1. *To construct MDS matrices from t-XOR matrices, we need to check only for t-XOR matrices whose nonzero elements (for A) are in the first t rows. This reduces the search space. For example, to construct recursive MDS matrices from* 2*-XOR matrices of order* 5 *and* 6*, we need to check only* $5! \times {}^8C_2 = 3360$ *and* $6! \times {}^{10}C_2 = 32400$ 2*-XOR matrices respectively.*

4.2.1 1-XOR matrices

We aim to minimize the number of nonzero elements in a matrix to obtain a recursive MDS matrix. It is important to note that to yield a recursive MDS matrix, an $n \times n$ matrix should have at least $n + 1$ nonzero elements. Therefore, we begin by focusing

on the 1-XOR matrices. Because of Lemma 4.1, we are considering those matrices whose first row has exactly two nonzero elements.

In Theorem 4.1, we prove that for $n \geq 3$, there exists no $n \times n$ 1-XOR matrix which is n-MDS. To proceed, we require the subsequent lemma.

Lemma 4.2. *Let $M = (P + A)$ be a 1-XOR matrix. Then, there exists some A_i's such that $M^r \leqq P^r + P^{r-1}A_1 + P^{r-2}A_2 + P^{r-3}A_3 + \cdots + PA_{r-1} + A_r$ for $1 \leq r \leq n$, where A_i are the matrices whose first row contain exactly i nonzero elements and rest rows zero.*

Proof. We will prove this result by mathematical induction. When $r = 1$, $M \leqq P + A_1 = P + PA_0 + A_1$. Hence, the statement holds for $r = 1$. Let us now assume that the statement is true for $r = k < n$. Next, we will show that the result is true for $r = k + 1$. Now $M^{k+1} = (P + A_1)^k(P + A_1)$. Therefore,

$$\begin{aligned} M^{k+1} &\leqq (P^k + P^{k-1}A_1 + P^{k-2}A_2 + P^{k-3}A_3 + \cdots + PA_{k-1} + A_k)(P + A_1) \\ &= (P^{k+1} + P^{k-1}A_1P + P^{k-2}A_2P + P^{k-3}A_3P + \cdots + PA_{k-1}P + A_kP) + \\ &\quad (P^kA_1 + P^{k-1}A_1^2 + P^{k-2}A_2A_1 + P^{k-3}A_3A_1 + \cdots + PA_{k-1}A_1 + A_kA_1). \end{aligned}$$

Note that $A_iA_1 \leqq A_1$ and $A_iP = A_i'$ for some A_i', where A_i' are the matrices whose first row contain exactly i nonzero elements and rest rows zero. Therefore,

$$\begin{aligned} M^{k+1} &\leqq (P^{k+1} + P^{k-1}A_1' + P^{k-2}A_2' + P^{k-3}A_3' + \cdots + PA_{k-1}' + A_k') + \\ &\quad (P^kA_1 + P^{k-1}A_1 + P^{k-2}A_1 + P^{k-3}A_1 + \cdots + PA_{k-1}A_1 + A_1) \\ &\leqq P^{k+1} + P^kA_1 + P^{k-1}(A_1' + A_1) + P^{k-2}(A_2' + A_1) \\ &\qquad + \cdots + P(A_{k-1}' + A_1) + (A_k' + A_1). \end{aligned}$$

Note that $A_i' + A_1 \leqq A_{i+1}''$, where A_i'' are the matrices whose first row contain exactly i nonzero elements and rest rows zero. Therefore,

$$M^{k+1} \leqq P^{k+1} + P^kA_1'' + P^{k-1}A_2'' + P^{k-2}A_3'' + \cdots + PA_k'' + A_{k+1}''.$$

Hence, the result. □

Theorem 4.1. *For $n \geq 3$, there does not exist any 1-XOR matrix of order n which is n-MDS.*

Proof. From Lemma 4.2, we have

$$\begin{aligned} M^n &\leqq (P+A)^n \\ &\leqq P^n + P^{n-1}A_1 + P^{n-2}A_2 + P^{n-3}A_3 + \cdots + PA_{n-1} + A_n. \end{aligned}$$

Thus,

$$\begin{aligned} |M^n| &\leq |P^n| + |P^{n-1}A_1| + |P^{n-2}A_2| + |P^{n-3}A_3| + \cdots + |PA_{n-1}| + |A_n| \\ &= |P^n| + |A_1| + |A_2| + |A_3| + \cdots + |A_{n-1}| + |A_n|. \end{aligned}$$

Note that P^n and A_n have a common element. Thus,

$$|M^n| \leq n + (1+2+3+\cdots+n) - 1 = \frac{n(n+3)}{2} - 1.$$

Therefore, for $n \geq 3$, $|M^n| < n^2$. Hence, the theorem. □

In the following theorem, we prove that some specific type of 1-XOR matrices of order n are not k-MDS for $k \leq 3n-5$.

Theorem 4.2. *For $n \geq 4$, let $M = PD + A$ be an $n \times n$ 1-XOR matrix over a field of characteristic 2, where P is a permutation matrix corresponding to an n length cycle permutation and A has a nonzero entry in $(1,1)$-th position. Then M is not k-MDS for $k \leq 3n-5$.*

Proof. Let P_i be the permutation matrix corresponding to the n length cycle permutation σ_i of the symmetric group S_n for i=$1, 2, \ldots, (n-1)!$. Suppose P_i and P_j be two permutation matrices corresponding to the n length cycle $\sigma_i = (i_1 = 1\ i_2\ i_3\ \ldots\ i_n)$ and $\sigma_j = (j_1 = 1\ j_2\ j_3\ \ldots\ j_n)$ respectively. Now consider the permutation $\lambda = \begin{pmatrix} j_1 & j_2 & j_3 & \cdots & j_n \\ i_1 & i_2 & i_3 & \ldots & i_n \end{pmatrix}$. Therefore, $\lambda\sigma_j\lambda^{-1} = (\lambda(j_1)\ \lambda(j_2)\ \lambda(j_3)\ \ldots\ \lambda(j_n)) = (i_1\ i_2\ i_3\ \ldots\ i_n) = \sigma_i$. Let Q be the permutation matrix corresponding to λ. Therefore, we have $QP_jQ^{-1} = P_i$. Hence, all P_i's for $i = 1, 2, \ldots, (n-1)!$ are permutation equivalent to each other. Therefore, for all $i, j \in \{1, 2, \ldots, (n-1)!\}$, $P_i = QP_jQ^{-1}$ for some permutation matrix Q, where the first row of Q is the first row the identity matrix.

By Lemma 2.10, we have $DQ^{-1} = Q^{-1}D'$, for some diagonal matrix D'. Also, since A has the nonzero entry in $(1,1)$-th position, we have

$$\begin{aligned} Q(P_jD + A)Q^{-1} &= QP_jQ^{-1}D' + A \\ &= P_iD' + A. \end{aligned}$$

Hence, all $P_iD + A$ for $i \in \{1, 2, \ldots, (n-1)!\}$ are permutation equivalent. Therefore, to check whether all such matrix $M = PD + A$, P is the permutation matrix corresponding to a full length cycle permutation and A has the nonzero entry in the

$(1,1)$-th position, is k-MDS, we only need to check for one such matrix M. Consider the matrix

$$M = \begin{bmatrix} a & 0 & 0 & \dots & 0 & x_n \\ x_1 & 0 & 0 & \dots & 0 & 0 \\ 0 & x_2 & 0 & \dots & 0 & 0 \\ 0 & 0 & x_3 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & x_{n-1} & 0 \end{bmatrix}.$$

Now consider the input vector $(0, 1, y, 0, \dots, 0)^T$. The resultant vector after each iteration is

$$\begin{bmatrix} 0 \\ 1 \\ y \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \xrightarrow[\text{i=1}]{} \begin{bmatrix} 0 \\ 0 \\ x_2 \\ x_3 y \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \xrightarrow[\text{i=2}]{} \begin{bmatrix} 0 \\ 0 \\ 0 \\ x_3 x_2 \\ x_4 x_3 y \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \xrightarrow[\text{i=3}]{} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ x_4 x_3 x_2 \\ x_5 x_4 x_3 y \\ \vdots \\ 0 \\ 0 \end{bmatrix} \xrightarrow[\text{i=4}]{} \cdots \xrightarrow[\text{i=n-5}]{} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \vdots \\ x_{n-4} x_{n-5} \cdots x_4 x_3 x_2 \\ x_{n-3} x_{n-4} x_{n-3} \cdots x_4 x_3 y \\ 0 \\ 0 \end{bmatrix} \xrightarrow[\text{i=n-4}]{} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \vdots \\ x_{n-3} x_{n-4} \cdots x_4 x_3 x_2 \\ x_{n-2} x_{n-3} \cdots x_4 x_3 y \\ 0 \end{bmatrix}$$

$$\xrightarrow[\text{i=n-3}]{} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \vdots \\ x_{n-2} x_{n-3} \cdots x_4 x_3 x_2 \\ x_{n-1} x_{n-2} x_{n-3} \cdots x_4 x_3 y \end{bmatrix} \xrightarrow[\text{i=n-2}]{} \begin{bmatrix} x_n x_{n-1} x_{n-2} x_{n-3} \cdots x_4 x_3 y \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ x_{n-1} x_{n-2} x_{n-3} \cdots x_4 x_3 x_2 \end{bmatrix} \xrightarrow[\text{i=n-1}]{} \begin{bmatrix} a x_n x_{n-1} x_{n-2} x_{n-3} \cdots x_4 x_3 y + \\ x_n x_{n-1} x_{n-2} x_{n-3} \cdots x_4 x_3 x_2 \\ * \\ 0 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ * \\ 0 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}$$

$$\xrightarrow[\text{i=n}]{\text{Let } y = a^{-1} x_2} \begin{bmatrix} 0 \\ 0 \\ * \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \cdots \xrightarrow[\text{i=2n-3}]{} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ * \end{bmatrix} \xrightarrow[\text{i=2n-2}]{} \begin{bmatrix} * \\ 0 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \xrightarrow[\text{i=2n-1}]{} \begin{bmatrix} * \\ * \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \xrightarrow[\text{i=2n}]{} \begin{bmatrix} * \\ * \\ * \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \cdots \xrightarrow[\text{i=2n+(n-5)=3n-5}]{} \begin{bmatrix} * \\ * \\ * \\ * \\ * \\ * \\ \vdots \\ * \\ 0 \\ 0 \end{bmatrix},$$

where $*$ denotes some entry may or may not be zero. The sum of nonzero elements of input vector and output vector in each iteration is $< n + 1$. Therefore, M is not k-MDS, where $k \leq 3n - 5$. □

Remark 4.2. *For $n = 3$, choose the input vector $y = (0, ax_2^{-1}, 1)^T$, and it can be easily checked that the sum of nonzero elements of input vector and output vector in $i = 2$, 3 and 4 is less than $n + 1$. Also, for the input vector $(0, 1, 0)^T$, the sum of nonzero elements of input vector and output vector in iteration $i = 1$ is less than $n + 1$. Therefore, the above result is also true for $n = 3$. Again for $n = 2$, the result*

is trivially true. Therefore, Theorem 4.2 is true for $n \geq 2$.

Remark 4.3. *For $n = 3$ and the input vector $y = (1, a^2x_2^{-1}x_3^{-1}, 0)$ it is easy to check that the above result holds for $i = 5, 6$. Therefore, for $n = 3$, Theorem 4.2 holds true for $k \leq 6$.*

Remark 4.4. *For $n = 4$ and the input vector $y = (0, a^2x_2^{-1}x_3^{-1}, 0, 1)$ it is easy check that the above result holds for $i = 1, 3, 4, 5, 6, 7, 8, 9, 10$. Therefore, for $n = 4$, Theorem 4.2 holds true for $k \leq 10$.*

Experimental observation: We have also observed that if in a 1-XOR matrix $M = PD + A$ of order n (for $n \leq 8$), if P is not a permutation matrix corresponding to a n length cycle permutation or A has a nonzero entry in some different position other than $(1, 1)$-th position, then M^k contains at least one zero entry for $k \leq 3n - 5$. Therefore, by Theorem 4.2, 1-XOR matrices of order n (for $n \leq 8$) are not k-MDS for $k \leq 3n - 5$ over a field of characteristic 2.

1-XOR matrix of order 2: There exists a 1-XOR matrix of order 2 which is 2-MDS. For example, consider the matrix

$$M = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \alpha & 0 \\ 0 & 1 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ \alpha & 0 \end{bmatrix}$$

over the field $\mathbb{F}_{2^4}$, where α is a primitive element with $\alpha^4 + \alpha + 1 = 0$. It can be verified that M^2 is an MDS matrix.

4.2.2 2-XOR matrices

Now we are considering two nonzero elements in A for the study of recursive MDS matrices.

2-XOR matrix of order 3: There exists a 2-XOR matrix of order 3 which is 3-MDS. For example, consider the matrix

$$M = \begin{bmatrix} 1 & 0 & 1 \\ 1 & \alpha & 0 \\ 0 & \alpha^3 + 1 & 0 \end{bmatrix}$$

over the field $\mathbb{F}_{2^4}$, where α is a primitive element with $\alpha^4 + \alpha + 1 = 0$. It is easy to check that M^3 is an MDS matrix.

2-XOR matrix of order 4: There exists a 2-XOR matrix of order 4 which is 4-MDS. For example, consider the matrix

$$M = \begin{bmatrix} 0 & 1 & \alpha^3 + 1 & 0 \\ 0 & 0 & \alpha & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

over the field $\mathbb{F}_{2^4}$, where α is a primitive element with $\alpha^4 + \alpha + 1 = 0$. It is easy to check that M^4 is an MDS matrix.

2-XOR matrix of order 5: We have observed that among 3360 2-XOR matrices of order 5 (see Remark 4.1), 12 matrices provide all nonzero element, when raised to power 5 but in a ring of characteristic 2 these matrices have zero elements when it raised to power 5. Therefore, over a ring of characteristic 2, there exist no 5-MDS 2-XOR matrix of order 5. Consequently, there are no 5-MDS 2-XOR matrices of order 5 over a ring of characteristic 2.

Thus, for 5×5 matrices, the minimum fixed XOR is 3 to obtain a 5-MDS matrix. In other words, to achieve a 5-MDS matrix, a minimum of 8 nonzero elements are needed in a 5×5 matrix.

2-XOR matrix of order 6: We have observed that among the 32400 2-XOR matrix of order 6, there exist no such matrix that gives all nonzero element when raised to power 6. Thus, for 6×6 matrices, the minimum fixed XOR is 3 to obtain a 6-MDS matrix. In other words for being 6-MDS, the minimum number of nonzero elements in a 6×6 matrix is $6 + 3 = 9$.

4.3 Study of DSI Matrices for the Construction of Recursive MDS Matrices

In the structure $M = PD_1 + D_2$, if

$$P = \begin{bmatrix} 0 & 0 & 0 & \dots & 0 & 1 \\ 1 & 0 & 0 & \dots & 0 & 0 \\ 0 & 1 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & 0 \\ 0 & 0 & 0 & \ddots & 1 & 0 \end{bmatrix}, \; D_2 = \begin{bmatrix} b_1 & 0 & 0 & \dots & 0 & 0 \\ 0 & b_2 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & b_{n-1} & 0 \\ 0 & 0 & 0 & \ddots & 0 & 0 \end{bmatrix} \tag{4.1}$$

and D_1 is a nonsingular diagonal matrix, then M is called a DSI matrix (see Definition 2.36).

In [TTKS18, Theorem 2], authors proved that given a DSI matrix M of order n, M^k contains at least one zero for $0 \leq k < n$ and $n \geq 2$. In Theorem 4.3, we prove this by providing a combinatorial argument. To proceed, we require the following lemma.

Lemma 4.3. *Let $M = P + D_2$ be an $n \times n$ matrix, where D_2 is a diagonal matrix having zero in the n-th diagonal position and P is the permutation matrix defined in 4.1. Then $M^r \leqq P^r + P^{r-1}D + P^{r-2}D + \cdots + PD + D_{n=0}$ for $r \geq 2$, where D denotes some nonsingular diagonal matrix and $D_{j=0}$ be some diagonal matrix with a 0 at the j-th diagonal position.*

Proof. We will prove this result using mathematical induction. We have

$$\begin{aligned} M^2 &= (P + D_2)(P + D_2) \\ &= P^2 + PD_2 + D_2P + D_2^2 \\ &\leqq P^2 + PD_{n=0} + D_{n=0}P + D_{n=0} \\ &\leqq P^2 + PD_{n=0} + PD_{n-1=0} + D_{n=0} \end{aligned}$$

Therefore,

$$M^2 \leqq P^2 + PD + D_{n=0}.$$

Hence, the statement holds for $r = 2$. Let us assume that the statement holds for $r = k$. Now, $M^{k+1} = M^k(P + D_2)$. Therefore, we have

$$\begin{aligned} M^{k+1} &\leqq (P^k + P^{k-1}D + P^{k-2}D + \cdots + PD + D_{n=0})(P + D_{n=0}) \\ &\leqq P^{k+1} + P^kDD_{n=0} + P^{k-1}DP + P^{k-1}DD_{n=0} + P^{k-2}DP + \\ &\qquad P^{k-2}DD_{n=0} + \cdots + PDP + PD_{n=0} + D_{n=0}P + D_{n=0} \\ &\leqq P^{k+1} + P^kD_{n=0} + P^kD + P^{k-1}D_{n=0} + P^{k-1}D + \cdots + \\ &\qquad + P^2D + PD_{n=0} + PD_{n-1=0} + D_{n=0} \\ &= P^{k+1} + P^k(D_{n=0} + D) + P^{k-1}(D_{n=0} + D) + \cdots + P(D_{n=0} + D_{n-1=0}) \\ &\qquad + D_{n=0} \\ &\leqq P^{k+1} + P^kD + P^{k-1}D + \cdots + PD + D_{n=0}. \end{aligned}$$

Thus, the lemma. □

Theorem 4.3. *Given a DSI matrix M of order $n \geq 2$, M^k is not MDS when $k < n$.*

Proof. From Lemma 4.3, we have

$$\begin{aligned} |M^k| &\leq |P^k D + P^{k-1} D + \cdots + PD + D_{n=0}| \\ &\leq |P^k D| + |P^{k-1} D| + \cdots + |PD| + |D_{n=0}|. \end{aligned}$$

Therefore, $|M^k| \leq \underbrace{|D| + |D| + \cdots + |D|}_{k \text{ times}} + |D_{n=0}| \leq kn + n - 1.$

Now for $k \leq n-1$, we have $|M^k| \leq (n-1)n + n - 1 = n^2 - 1$. Hence, the result. □

Recall that in the DSI matrix structure $M = PD_1 + D_2$ of order n, if $D_2 = diag(b_1, 0, b_3, \ldots, 0, b_{n-1}, 0)$ (when n is even) or $D_2 = diag(b_1, 0, b_3, \ldots, b_{n-2}, b_{n-1}, 0)$ (when n is odd), then M is called a sparse DSI matrix of order n.

Lemma 4.4. *Suppose $n \geq 2$. Let $M = P + D_2$ be an $n \times n$ matrix, where D_2 have zeros in the i-th and $(i \ mod \ n+1)$-th diagonal position and P is the permutation matrix defined in 4.1. Then $M^r \leqq P^r + P^{r-1} D + P^{r-2} D + \cdots + PD_{i=0} + D_{i,i+1=0}$ for $r \geq 2$, where D denotes some nonsingular diagonal matrix and $D_{j,k=0}$ be some diagonal matrix with 0 at the j-th and k-th diagonal positions.*

Proof. We will prove this result using mathematical induction. We simply denote $(i+1)$ for $(i \ mod \ n+1)$. We have

$$\begin{aligned} M^2 &= (P + D_2)(P + D_2) \\ &= P^2 + PD_2 + D_2 P + D_2^2 \\ &\leqq P^2 + PD_{i,i+1=0} + PD_{i-1,i=0} + D_{i,i+1=0} \\ &= P^2 + P(D_{i,i+1=0} + D_{i-1,i=0}) + D_{i,i+1=0}. \end{aligned}$$

Therefore,

$$M^2 \leqq P^2 + PD_{i=0} + D_{i,i+1=0}.$$

Hence, the statement is valid for $r = 2$. Let us assume that the statement holds for $r = k$.

Now, $M^{k+1} = M^k(P + D_2)$. Therefore, we obtain the following:

$$\begin{aligned} M^{k+1} &\leqq (P^k + P^{k-1} D + P^{k-2} D + \cdots + PD_{i=0} + D_{i,i+1=0})(P + D_{i,i+1=0}) \\ &\leqq P^{k+1} + P^k D_{i,i+1=0} + P^{k-1} DP + P^{k-1} DD_{i,i+1=0} + \cdots + \\ &\qquad + PD_{i=0} P + PD_{i=0} D_{i,i+1=0} + D_{i,i+1=0} P + D_{i,i+1=0} \\ &\leqq P^{k+1} + P^k D_{i,i+1=0} + P^k D + P^{k-1} D_{i,i+1=0} + \cdots + \\ &\qquad + P^2 D_{i-1=0} + PD_{i,i+1=0} + PD_{i-1,i=0} + D_{i,i+1=0} \end{aligned}$$

$$= P^{k+1} + P^k(D_{i,i+1=0} + D) + \cdots + P(D_{i,i+1=0} + D_{i-1,i=0}) + D_{i,i+1=0}$$
$$\leqq P^{k+1} + P^kD + \cdots + PD_{i=0} + D_{i,i+1=0}.$$

Thus, the lemma. □

Theorem 4.4. *Let $M = PD_1 + D_2$ be an $n \times n$ matrix, where P be the permutation matrix defined in 4.1, D_1 is a nonsingular diagonal matrix and D_2 has any two consecutive zero entries in the diagonal position, then M^k must contain a zero entry for $2 \leq k \leq n$.*

Proof. From Lemma 4.4, we have

$$M^k \leqq (P + D_2)^k$$
$$\leqq P^k + P^{k-1}D + P^{k-2}D + \cdots + PD_{i=0} + D_{i,i+1=0}.$$

Therefore, we have

$$|M^k| \leq |P^k + P^{k-1}D + P^{k-2}D + \cdots + PD_{i=0} + D_{i,i+1=0}|$$
$$\leq |P^k| + |P^{k-1}D| + |P^{k-2}D| + \cdots + |PD_{i=0}| + |D_{i,i+1=0}|$$
$$\leq (k-1)n + (n-1) + (n-2)$$
$$= (k-1)n + 2n - 3.$$

When $k \leq n-1$, we have $|M^k| \leq (n-2)n + 2n - 3 = n^2 - 3 < n^2$. It is easy to check that $P^n = I$, where I is the identity matrix. Thus, for $k = n$, we have

$$|M^n| \leq |I + P^{n-1}D + P^{n-2}D + \cdots + PD_{i=0} + D_{i,i+1=0}|$$
$$\leq |I + D_{i,i+1=0}| + |P^{n-1}D| + |P^{n-2}D| + \cdots + |PD_{i=0}|.$$

Note that $|I + D_{i,i+1=0}| = n$ and $|PD_{i=0}| = n - 1$. Hence,

$$|M^n| \leq \underbrace{n + n + \cdots + n}_{n-1 \text{ times}} + n - 1 = (n-1)n + (n-1) = n^2 - 1 < n^2.$$

Hence, the result. □

The authors in [TTKS18], could not find a sparse DSI matrix of order 8 which is 8-MDS, over the field $\mathbb{F}_{2^8}$ due to large search space. In the following lemma and theorem, we have provided an equivalence criteria for checking a $n \times n$ sparse DSI matrix to be a n-MDS. Through these results, we reduce the large search space into a small search space and show that there exists no 8×8 sparse DSI matrix over $\mathbb{F}_{2^8}$ which is 8-MDS.

Lemma 4.5. *Let $a \in \mathbb{F}_q^*$ and P is an $n \times n$ permutation matrix. Given any $n \times n$ diagonal matrix D, there exists an $n \times n$ diagonal matrix D' such that $(P + D)^k$ is MDS if and only if $(aP + D')^k$ is MDS for $k \geq 1$.*

Proof. Note that $(P + D)^k$ is MDS if and only if $a^k(P + D)^k$ because $a \neq 0$. Now we have

$$\begin{aligned} a^k(P + D)^k &= (a(P + D))^k \\ &= (aP + aD)^k = (aP + D')^k, \end{aligned}$$

where $D' = aD$. □

Note that in the above lemma as $D' = aD$, D and D' have nonzeros in the same position.

Theorem 4.5. *Let $a_1, a_2, \ldots, a_n \in \mathbb{F}_q^*$, $D_1 = diag(a_1, a_2, \ldots, a_n)$ and*

$$P = \begin{bmatrix} 0 & 0 & 0 & \ldots & 0 & 1 \\ 1 & 0 & 0 & \ldots & 0 & 0 \\ 0 & 1 & 0 & \ldots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \ddots & 1 & 0 \end{bmatrix}.$$

Given any diagonal matrix D_2, there exists $a \in \bar{\mathbb{F}}_q$ satisfying $a^n = a_1 a_2 \ldots a_n$ such that $(PD_1 + D_2)^k$ is MDS if and only if $(aP + D')^k$ is MDS for $r \geq 1$.

Proof. Consider a nonsingular diagonal matrix $D_d = diag(d_1, d_2, \ldots, d_n)$. Let $D_{d,a} = diag(ad_2^{-1}, ad_3^{-1}, \ldots, ad_n^{-1}, ad_1^{-1})$. Then, we have

$$\begin{aligned} aP + D_2 &= aD_dD_d^{-1}PD_dD_d^{-1} + D_dD_2D_d^{-1} \\ &= D_d(aD_d^{-1}PD_d + D_2)D_d^{-1} \\ &= D_d(PD_{d,a}D_d + D_2)D_d^{-1} \\ &= D_d(PD_1 + D_2)D_d^{-1}, \end{aligned} \tag{4.2}$$

where $D_1 = D_{d,a}D_d$.

Now we will show that there exists $D_{d,a}$ such that $D_1 = D_{d,a}D_d$. If $D_1 = D_{d,a}D_d$,

we have

$$\begin{aligned}
a_1 &= ad_2^{-1}d_1 \\
a_2 &= ad_3^{-1}d_2 \\
a_3 &= ad_4^{-1}d_3 \\
&\dots \\
a_{n-1} &= ad_n^{-1}d_{n-1} \\
a_n &= ad_1^{-1}d_n.
\end{aligned}$$

Therefore, we have

$$\begin{aligned}
d_2 &= aa_1^{-1}d_1 \\
d_3 &= a^2a_1^{-1}a_2^{-1}d_1 \\
d_4 &= a^3a_1^{-1}a_2^{-1}a_3^{-1}d_1 \\
&\dots \\
d_{n-1} &= a^{n-2}a_1^{-1}a_2^{-1}a_3^{-1}\dots a_{n-2}^{-1}d_1 \\
d_n &= a^{n-1}a_1^{-1}a_2^{-1}a_3^{-1}\dots a_{n-2}^{-1}a_{n-1}^{-1}d_1 \\
d_1 &= a^n(a_1^{-1}a_2^{-1}a_3^{-1}\dots a_{n-2}^{-1}a_n^{-1})d_1.
\end{aligned}$$

Thus, $a^n = a_1a_2a_3\dots a_n$, for $a_1, a_2, \dots, a_n \in \mathbb{F}_q^*$ and such a exists in $\bar{\mathbb{F}}_q$. Therefore, from Equation 4.2, we can say that $(PD_1 + D_2)^k$ is MDS if and only if $(aP + D_2)^k$ is MDS. □

Corollary 4.1. *Let $a_i \in \mathbb{F}_q^*$ for $1 \le i \le n$ and $a \in \bar{\mathbb{F}}_q$ satisfying $a^n = a_1a_2\dots a_n$. Let $b_j \in \mathbb{F}_q$ and $b'_j = a^{-1}b_j$ for $1 \le j \le n-1$. Suppose $M = DSI(a_1, a_2, \dots, a_n; b_1, b_2, \dots, b_{n-1})$ and $M_1 = DSI(1, 1, \dots, 1; b'_1, b'_2, \dots, b'_{n-1})$. Then the matrix M^k is MDS if and only if the matrix M_1^k is MDS for $r \ge 1$.*

Proof. Let $M_2 = DSI(a, a, \dots, a; b_1, b_2, \dots, b_{n-1})$. From Theorem 4.5, M^k is MDS if and only if M_2^k is MDS for $r \ge 1$. From Lemma 4.5, M_2^k is MDS if and only if M_1^k is MDS. Hence, the corollary. □

4.3.1 Non-existence of 8-MDS sparse DSI matrix of order 8 over $\mathbb{F}_{2^8}$

We will now prove that there does not exist any 8-MDS sparse DSI matrix of order 8 over the field $\mathbb{F}_{2^8}$. From Corollary 4.1, any sparse DSI matrix $M_1 = DSI(a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8; b'_1, 0, b'_3, 0, b'_5, 0, b'_7, 0)$ over $\mathbb{F}_{2^8}$ is permutation equivalent to a sparse DSI matrix $M = DSI(1, 1, 1, 1, 1, 1, 1, 1; b_1, 0, b_3, 0, b_5, 0, b_7, 0)$ where $b_i = a^{-1}b'_i$ for $i = 1, 3, 5, 7$ and $a^n = a_1a_2\dots a_8$.

Since $x \to x^8$ is an isomorphism over $\mathbb{F}_{2^8}$, such a exists in the field $\mathbb{F}_{2^8}$. Therefore, it is sufficient to check only those M whose b_i's belong to $\mathbb{F}_{2^8}$ which has $(2^8)^4 = 2^{32}$ choices. Otherwise, the total choices would be $(2^8)^{12} = 2^{96}$ considering $a_1, a_2, \cdots, a_8, b'_1, b'_3, b'_5, b'_7$ all belong to $\mathbb{F}_{2^8}$. This was perhaps the reason why the authors in [TTKS18] could not provide the answer for either the possibility or impossibility of 8-MDS sparse DSI matrix of order 8 over $\mathbb{F}_{2^8}$.

After reducing the search space from 2^{96} candidates to 2^{32} candidates only, we experimentally observed that M^8 over a field of characteristic 2 will not be MDS if the following conditions are satisfied.

1. If $b_1 + b_3 + b_5 + b_7 = 0$
2. If $b_1b_3 + b_1b_5 + b_1b_7 + b_3b_5 + b_3b_7 + b_5b_7 = 0$
3. If $b_1b_3b_5 + b_1b_3b_7 + b_1b_5b_7 + b_3b_5b_7 = 0$
4. If $b_1b_3b_5b_7 = 0$
5. If $b_1 = b_3$ or $b_1 = b_7$ or $b_3 = b_5$ or $b_5 = b_7$.

One can get the above conditions by (i) looking at some of the entries of M^8 and M^{-8} and (ii) computing the determinants of some of the 2×2 matrices in M^8 and M^{-8}. We want to emphasize that these are not the only conditions we got from (i) and (ii); these are only a few. One can get many more such conditions and can further enhance the search time. We considered only five because the first four conditions are symmetric in b_1, b_3, b_5 and b_7 and the fifth one appears very simple.

We ran an experiment over all choices of $b_1, b_3, b_5, b_7 \in \mathbb{F}_{2^8}$ except which satisfy at least one of the above five conditions. Our experiment could not find any 8-MDS matrix of order 8. Thus, we conclude that there does not exist any sparse DSI matrix of order 8 over the field $\mathbb{F}_{2^8}$ which is 8-MDS.

Remark 4.5. *An 8-MDS sparse DSI matrix of order 8 exists over the higher order field. For example, consider the matrix*

$$M = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & \alpha & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & \alpha^{12} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & \alpha^{30} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \alpha^{30} & 0 \end{bmatrix}$$

over $\mathbb{F}_{2^{10}}$*, where* α *is a root of the constructing polynomial* $x^{10} + x^3 + 1$*. It can be verified that* M^8 *is MDS.*

4.4 Conclusion

This chapter provides a systematic study of low fixed XOR matrices, including DSI and sparse DSI matrices, and provides some impossibility results on the construction of recursive MDS matrices from them. Our investigations in this chapter open up possibilities for future work, such as:

1. Although we have provided an upper bound on the number of nonzero elements in 1-XOR matrices of order n when raised to power n, we have not found similar results for t-XOR matrices with $t > 1$. Therefore, future work can explore finding an upper bound on the number of nonzero elements in t-XOR matrices of order n when raised to power n.
2. There are many direct constructions of recursive MDS matrices from companion matrices. So it can be a problem for further research to find a direct construction of recursive MDS matrices from sparse DSI matrices.

5

Design and Analysis of Recursive MDS Matrices Using DLS Matrices

Contents

5.1 Introduction

Efficient implementation is crucial for lightweight cryptographic primitives, but strong diffusion power is also essential for security. This often means a trade-off between the two. To address this challenge, recursive MDS matrices have been proposed. A matrix B is considered a recursive MDS matrix if the matrix B^q is MDS for some positive integer q. The implementation of B^q can be accomplished by iteratively executing the implementation of B, necessitating q clock cycles. One advantageous aspect of such matrices is their suitability for lightweight implementations, as the hardware cost relies on the matrix B rather than the MDS matrix B^q.

The utilization of such matrices derived from companion matrices has been observed in the PHOTON hash function family [GPP11] and the LED block cipher [GPPR11]. Later on, several sparse matrix structures are proposed, in-

cluding Generalized-Feistel-Structure (GFS) [WWW13], Diagonal-Serial-Invertible (DSI) [TTKS18] and sparse DSI [TTKS18] matrices.

This chapter introduces a new class of sparse matrices called Diagonal-like sparse (DLS) matrices. It establishes that an n-MDS DLS matrix of order n requires a fixed XOR value ($\mathcal{K}$) of at least $\lceil \frac{n}{2} \rceil$. Furthermore, it demonstrates that an n-MDS DLS matrix over $\mathbb{F}_{2^r}$ with $\mathcal{K} = \lceil \frac{n}{2} \rceil$ is a permutation similar to some n-MDS sparse DSI matrix. This implies that the existence of n-MDS DLS matrices with the lowest fixed XOR value is equivalent to the existence of n-MDS sparse DSI matrices over $\mathbb{F}_{2^r}$ and vice versa.

In addition, this chapter introduces another class of sparse matrices called generalized DLS (GDLS) matrices that generalize the structure of DLS matrices. Using these matrices, the chapter proposes some lightweight recursive MDS matrices of orders $4, 5, 6$, and 7, which can be implemented with $22, 30, 31$, and 45 XORs over $\mathbb{F}_{2^8}$, respectively. The results match the best known lightweight recursive MDS matrices for orders 4 and 6, and outperform the best known matrices for orders 5 and 7. Additionally, a 4-MDS GDLS matrix over $\mathbb{F}_{2^4}$ with a XOR count of 10 is proposed, which meets the best known result. In addition to searching over $\mathbb{F}_{2^4}$ and $\mathbb{F}_{2^8}$, the chapter provides some efficient n-MDS GDLS matrices over $GL(8, \mathbb{F}_2)$ for orders $n = 4, 5$, and 6. Table 5.1 compares the presented results with the known results.

Outline: The rest of this chapter is structured as follows: Section 5.2 discusses DLS matrices and provides theoretical results to limit the search space for finding n-MDS DLS matrices of order n. Section 5.3 proposes some lightweight recursive MDS matrices of orders $4, 5, 6$, and 7 using GDLS matrices. Finally, Section 5.4 concludes the chapter and explores potential avenues for future research.

5.2 Construction of Recursive MDS Matrices from DLS Matrices

An MDS matrix must have all its entries nonzero. Therefore, any $n \times n$ matrix cannot be MDS if the number of nonzero entries is less than n^2. In this section, we are using this fact to obtain some interesting results.

Table 5.1: Comparison of n-MDS matrices of order n.

Order n	Input	Matrix Type	Field/Ring	XOR	References
4	4-bit	Sparse DSI	$\mathbb{F}_{2^4}$/0x13	10	[TTKS18]
4	4-bit	GFS	$GL(4, \mathbb{F}_2)$	10	[WWW13]
4	4-bit	Companion	$\mathbb{F}_{2^4}$/0x13	15	[KPPY14]
4	4-bit	Companion	$GL(4, \mathbb{F}_2)$	15	[WWW13]
4	4-bit	GDLS	$\mathbb{F}_{2^4}$/0x13	10	Section 5.3.1
4	8-bit	Sparse DSI	$[\mathbb{F}_{2^4}/0x13]^2$	2×10	[TTKS18]
4	8-bit	GFS	$GL(8, \mathbb{F}_2)$	18	[WWW13]
4	8-bit	Companion	$\mathbb{F}_{2^4}$/0x11d	33	[KPPY14]
4	8-bit	Companion	$GL(8, \mathbb{F}_2)$	27	[WWW13]
4	8-bit	Sparse DSI	$GL(8, \mathbb{F}_2)$	18	[LSS+20]
4	8-bit	GDLS	$\mathbb{F}_{2^8}$/0x1c3	22	Remark 5.10
4	8-bit	GDLS	$GL(8, \mathbb{F}_2)$	18	Example 5.2
5	4-bit	Companion	$GL(4, \mathbb{F}_2)$	19	[WWW13]
5	4-bit	Companion	$\mathbb{F}_{2^4}$/0x13	18	[GPP11]
5	4-bit	Companion	$\mathbb{F}_{2^4}$/0x13	18	[TTKS18]
5	4-bit	GDLS	$\mathbb{F}_{2^4}$/0x13	26	Section 5.3.2
5	8-bit	Companion	$GL(8, \mathbb{F}_2)$	35	[WWW13]
5	8-bit	Sparse DSI	$\mathbb{F}_{2^8}$/0x1c3	31	[TTKS18]
5	8-bit	Sparse DSI	$GL(8, \mathbb{F}_2)$	30	[LSS+20]
5	8-bit	GDLS	$\mathbb{F}_{2^8}$/0x1c3	30	Section 5.3.2
5	8-bit	GDLS	$GL(8, \mathbb{F}_2)$	**28**	Remark 5.12
6	4-bit	Companion	$GL(4, \mathbb{F}_2)$	25	[WWW13]
6	4-bit	Companion	$\mathbb{F}_{2^4}$/0x13	28	[GPP11]
6	4-bit	Companion	$\mathbb{F}_{2^4}$/0x13	25	[TTKS18]
6	8-bit	Companion	$GL(8, \mathbb{F}_2)$	45	[WWW13]
6	8-bit	Companion	$\mathbb{F}_{2^8}$/0x11b	57	[GPP11]
6	8-bit	Sparse DSI	$\mathbb{F}_{2^8}$/0x1c3	31	[TTKS18]
6	8-bit	GDLS	$\mathbb{F}_{2^8}$/0x1c3	31	Section 5.3.3
6	8-bit	GDLS	$GL(8, \mathbb{F}_2)$	**30**	Remark 5.13
7	4-bit	Companion	$GL(4, \mathbb{F}_2)$	30	[WWW13]
7	4-bit	Companion	$\mathbb{F}_{2^4}$/0x13	31	[GPP11]
7	4-bit	Companion	$\mathbb{F}_{2^4}$/0x13	30	[TTKS18]
7	8-bit	Companion	$GL(8, \mathbb{F}_2)$	54	[WWW13]
7	8-bit	Sparse DSI	$\mathbb{F}_{2^8}$/0x1c3	54	[TTKS18]
7	8-bit	Sparse DSI	$\mathbb{F}_{2^8}$/0x1c3	47	[KSV19]
7	8-bit	GDLS	$\mathbb{F}_{2^8}$/0x1c3	**45**	Section 5.3.4
8	4-bit	Companion	$GL(4, \mathbb{F}_2)$	37	[WWW13]
8	4-bit	Companion	$\mathbb{F}_{2^4}$/0x13	47	[GPP11]
8	4-bit	Companion	$\mathbb{F}_{2^4}$/0x13	36	[TTKS18]
8	4-bit	Companion	$\mathbb{F}_{2^4}$/0x13	41	[KPPY14]
8	8-bit	Companion	$GL(8, \mathbb{F}_2)$	65	[WWW13]
8	8-bit	Companion	$[\mathbb{F}_{2^4}/0x13]^2$	2×36	[TTKS18]

Definition 5.1. *(DLS matrix) Let* $\rho = [i_1, i_2, i_3, \ldots, i_n]$ *be a permutation such that* $i_k \neq k$ *for* $k = 1, 2, \ldots, n$, D_1 *be a nonsingular diagonal matrix and* D_2 *be a diagonal matrix (may be singular). Then we will call the matrix*

$$B = PD_1 + D_2$$

as the diagonal-like sparse (DLS) matrix, where P *is the permutation matrix of order* n *related to the permutation* ρ*. The matrices denoted as* $DLS(\rho; D_1, D_2)$*.*

Example 5.1. *An example of a DLS matrix of order* 4 *is given by*

$$DLS(\rho; D_1, D_2) = PD_1 + D_2 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} a & 0 & 0 & 0 \\ 0 & b & 0 & 0 \\ 0 & 0 & c & 0 \\ 0 & 0 & 0 & d \end{bmatrix} + \begin{bmatrix} e & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & f & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$
$$= \begin{bmatrix} e & 0 & 0 & d \\ 0 & 0 & c & 0 \\ a & 0 & f & 0 \\ 0 & b & 0 & 0 \end{bmatrix},$$

where P *is the permutation matrix related to* $\rho = [3, 4, 2, 1]$ *and* D_1*=*$diag(a, b, c, d)$ *and* $D_2 = diag(e, 0, f, 0)$*.*

Remark 5.1. *Note that the DSI matrix, as defined in Definition 2.36, is a particular type of DLS matrix. More specifically, for* $\rho = [2, 3, 4, \ldots, n-1, n, 1]$ *and a nonsingular diagonal matrix* D_1*, if* $D_2 = diag(b_1, b_2, \ldots, b_{n-1}, 0)$ *then we call* $DLS(\rho; D_1, D_2)$ *a DSI matrix and if* $D_2 = diag(b_1, 0, b_3, \ldots, 0, b_{n-1}, 0)$ *(when* n *is even) or* $D_2 = diag(b_1, 0, b_3, \ldots, b_{n-2}, b_{n-1}, 0)$ *(when* n *is odd), then* $DLS(\rho; D_1, D_2)$ *is called a sparse DSI matrix of order* n*.*

In Theorem 5.1, we discuss the minimum power required for a DLS matrix of order n to be an MDS. To establish this result, we require the following lemma. The proof of the lemma will follow a similar approach to the proof of Lemma 4.3. For brevity, we state the result without providing a proof.

Lemma 5.1. *Let* $M = P + D_2$ *be an* $n \times n$ *matrix, where* D_2 *is a diagonal matrix (may be singular) and* P *is a permutation matrix. Then*

$$M^r \leqq P^r + P^{r-1}D + P^{r-2}D + \ldots + PD + D_2^2$$

for $r \geq 2$, where D denotes some nonsingular diagonal matrix.

Theorem 5.1. *Given a DLS matrix $M = DLS(\rho; D_1, D_2)$ of order $n \geq 2$, for $k < n-1$, the number of nonzero elements in M^k is less than n^2 and hence M^k is not MDS.*

Proof. We have $M = DLS(\rho; D_1, D_2) \leqq P + D_2$, where P is the permutation matrix corresponding to ρ. From Lemma 5.1, we have

$$\begin{aligned} |M^k| &\leq |P^k + P^{k-1}D + \ldots + PD + D_2^2| \\ &\leq |P^k D| + |P^{k-1}D| + \ldots + |PD| + D_2^2|. \end{aligned} \tag{5.1}$$

Since power of a permutation matrix is again a permutation matrix, we have

$$|M^k| \leq \underbrace{|D| + |D| + \ldots + |D|}_{k \text{ times}} + |D_2^2| \leq kn + n. \tag{5.2}$$

Now for $k < n-1$, we have

$$|M^k| < (n-1)n + n = n^2 \implies |M^k| < n^2.$$

Hence, M^k is not MDS for $k < n-1$. □

Remark 5.2. *From the above theorem, we know that for a DLS matrix $M = DLS(\rho; D_1, D_2)$ of order $n \geq 2$, M^k is not an MDS for $k < n-1$. However, there exist k-MDS DLS matrix for $k = n-1$. For example, consider the DLS matrix $M = DLS(\rho; D_1, D_2)$ of order 4 with $\rho = [4, 1, 2, 3]$, $D_1 = diag(\alpha^2, \alpha^2, \alpha^2, 1)$ and $D_2 = diag(\alpha^2, 1, \alpha^2, 1)$, where α is a primitive element of $\mathbb{F}_{2^4}$ with $\alpha^4 + \alpha + 1 = 0$. It can be checked that the matrix*

$$\begin{aligned} M &= DLS(\rho; D_1, D_2) \\ &= \begin{bmatrix} \alpha^2 & \alpha^2 & 0 & 0 \\ 0 & 1 & \alpha^2 & 0 \\ 0 & 0 & \alpha^2 & 1 \\ \alpha^2 & 0 & 0 & 1 \end{bmatrix} \end{aligned}$$

is 3-MDS.

We will now examine the influence of the permutation ρ on a DLS matrix $DLS(\rho; D_1, D_2)$ in the construction of recursive MDS matrices. To accomplish this, we will utilize the following lemma.

Lemma 5.2. *If ρ is not an n-cycle for a DLS matrix $M = DLS(\rho; D_1, D_2)$ of order $n \geq 2$, then M^{n-1} and M^n contain at most $n^2 - 2n$ and $n^2 - n - 2$ nonzero elements respectively.*

Proof. If ρ is not an n-cycle, then it is a product of disjoint cycles in S_n. Suppose that $\rho = \rho_1\rho_2 \dots \rho_v$, where ρ_i is a r_i-cycle in S_n for $i = 1, 2, \dots, v$ and $v \in \{2, 3, \dots, \lfloor \frac{n}{2} \rfloor\}$. But by the definition of the DLS matrix, ρ has no fixed points, we have $2 \leq r_i \leq n-2$ and $r_1 + r_2 + \cdots + r_v = n$.

Now from Equation 6.2, we have $|M^{n-1}| \leq n^2$ and $|M^n| \leq n^2 + n$. However, we can eliminate some counting of nonzero elements based on the following conditions:

1. For the permutation matrix P related to ρ, $P^{r_i}D$ has r_i nonzero elements in the diagonal. Also, D has n nonzero elements in the diagonal.
2. $P^{r_i+1}D$ and PD have r_i nonzero elements in the same positions.
3. Since $v \leq \lfloor \frac{n}{2} \rfloor$, at least two multiples of some r_i occurs in the set $\{2, 3, \cdots, n\}$. Thus, $P^{r_i}D$ and $P^{2r_i}D$ have at least r_i nonzero elements in the same diagonal position.

Therefore, we have

$$|M^{n-1}| \leq n^2 - 2 \cdot (r_1 + r_2 + \cdots + r_v) \leq n^2 - 2n$$
$$\text{and } |M^n| \leq n^2 + n - 2 \cdot (r_1 + r_2 + \cdots + r_v) - r_i \leq n^2 - n - 2.$$

Hence, the result. □

Corollary 5.1. *For a DLS matrix $M = DLS(\rho; D_1, D_2)$ of order $n \geq 2$, if ρ is not an n-cycle, then M^k is not MDS for $k \leq n$.*

Remark 5.3. *If ρ is not an n-cycle, then by the Condition 1 of the above proof and Equation 5.1, we can say $|M^{n-2}| < n^2 - n$.*

To this point, we have ignored the possibility that the diagonal of D_2 contains zero entries. We now look at the case in which D_2 is singular, i.e., its diagonal contains at least one zero.

Lemma 5.3. *In a DLS matrix $M = DLS(\rho_1; D_1, D_2)$ of order $n \geq 2$, if D_2 is singular, then M^k cannot be MDS for $k \leq n - 1$, even if ρ is n-cycle.*

Proof. If D_2 is singular, having at least one zero in the diagonal then from Equation 5.1 we have,

$$\begin{aligned}|M^k| &\leq |P^kD| + |P^{k-1}D| + \ldots + |PD| + |D_{i=0}| \\ &\leq \underbrace{n + n + \ldots + n}_{k \text{ times}} + (n-1) = kn + n - 1.\end{aligned}$$

Where $D_{i=0}$ be some diagonal matrix with a zero at the i-th diagonal position for some $i \in \{1, 2, \ldots, n\}$. Thus, for $k \leq n-1$, we have $|M^k| \leq n^2 - 1$. Therefore, if D_2 is singular, a DLS matrix of order $n \geq 2$, cannot be k-MDS for $k \leq n-1$. □

Remark 5.4. *When D_2 is singular, a DLS matrix $M = DLS(\rho; D_1, D_2)$ of order $n \geq 2$, can be a k-MDS for $k = n$. For example, consider the DLS matrix $M = DLS(\rho; D_1, D_2)$ of order 4 with $\rho = [4, 1, 2, 3]$, $D_1 = diag(\alpha^2, \alpha^2, \alpha^2, 1)$ and $D_2 = diag(1, 0, \alpha, 0)$, where α is a primitive element of $\mathbb{F}_{2^4}$ with $\alpha^4 + \alpha + 1 = 0$. It can be verified that the matrix*

$$\begin{aligned}M &= DLS(\rho; D_1, D_2) \\ &= \begin{bmatrix} 1 & \alpha^2 & 0 & 0 \\ 0 & 0 & \alpha^2 & 0 \\ 0 & 0 & \alpha & 1 \\ \alpha^2 & 0 & 0 & 0 \end{bmatrix}\end{aligned}$$

is 4-MDS.

If D_2 is nonsingular, then the fixed XOR count (see Section 2.6.1) for a DLS matrix is n. Whereas for the companion matrix, DSI matrix and sparse DSI matrix, the fixed XOR is $n-1$, $n-1$, and $\lceil \frac{n}{2} \rceil$ respectively. However, we can reduce the number of nonzero elements for D_2 in the DLS matrices to get a recursive MDS matrix from this. In this context, we have proved in Theorem 5.2 that in an n-MDS DLS matrix $DLS(\rho; D_1, D_2)$ of order n, D_2 must have at least $\lceil \frac{n}{2} \rceil$ nonzero elements. Thus, for an n-MDS DLS matrix of order n, the fixed XOR can be reduced to $\lceil \frac{n}{2} \rceil$.

Lemma 5.4. *Let $M = P + D_2$ be an $n \times n$ matrix, where D_2 is a diagonal matrix having at most $t = \lceil \frac{n}{2} \rceil - 1$ nonzero elements and P is permutation matrix. Then*

$$M^r \leqq P^r + P^{r-1}D + P^{r-2}D + \ldots + PD_{i=0} + D_{\{t\}=0}$$

for $r \geq 2$, where D denotes some nonsingular diagonal matrix, $D_{i=0}$ be some diagonal matrix with a zero at the i-th diagonal position for some $i \in \{1, 2, \ldots, n\}$ and $D_{\{t\}=0}$ be some diagonal matrix with t many zeros in the diagonal.

Proof. We will prove this result using mathematical induction. We have

$$M^2 = (P + D_2)(P + D_2) = P^2 + PD_2 + D_2P + D_2^2.$$

By Lemma 2.10, we have $D_{\{t\}=0}P = PD'_{\{t\}=0}$ for some diagonal matrix $D'_{\{t\}=0}$ with t many zeros in the diagonal and so $D_{\{t\}=0} + D'_{\{t\}=0}$ has at least one zero in the diagonal. Therefore,

$$M^2 \leqq P^2 + PD_{\{t\}=0} + PD'_{\{t\}=0} + D_{\{t\}=0} \leqq P^2 + PD_{i=0} + D_{\{t\}=0}.$$

Therefore, the statement holds for $r = 2$. Let us assume that the statement is true for $r = k$.
Now, $M^{k+1} = M^k(P + D_2)$. Therefore, we have

$$\begin{aligned} M^{k+1} &\leqq (P^k + P^{k-1}D + P^{k-2}D + \ldots + PD_{i=0} + D_{\{t\}=0})(P + D_{\{t\}=0}) \\ &\leqq P^{k+1} + P^kD_{\{t\}=0} + P^{k-1}DP + P^{k-1}DD_{\{t\}=0} + \ldots + \\ &\quad + PD_{i=0}P + PD_{i=0}D_{\{t\}=0} + D_{\{t\}=0}P + D_{\{t\}=0} \\ &\leqq P^{k+1} + P^kD_{\{t\}=0} + P^kD + P^{k-1}D_{\{t\}=0} + \ldots + P^2D'_{j=0} \\ &\quad + PD_{\{t\}=0} + PD'_{\{t\}=0} + D_{\{t\}=0}, \end{aligned}$$

where $D_{i=0}P = PD'_{j=0}$ for some $D'_{j=0}$ and $D_{\{t\}=0}P = PD'_{\{t\}=0}$ for some diagonal matrix $D'_{\{t\}=0}$ with t many zeros in the diagonal. Thus, we have

$$\begin{aligned} M^{k+1} &\leqq P^{k+1} + P^k(D_{\{t\}=0} + D) + \ldots + P(D_{\{t\}=0} + D'_{\{t\}=0}) + D_{\{t\}=0} \\ &\leqq P^{k+1} + P^kD + \ldots + PD_{i=0} + D_{\{t\}=0}. \end{aligned}$$

This completes the proof of the lemma. □

Remark 5.5. *If D_2 has $t = \lceil \frac{n}{2} \rceil$ nonzero elements then $D_{\{t\}=0} + D'_{\{t\}=0}$ may be nonsingular. For example, let $\rho = [2, 4, 1, 3]$ and $D = diag(a_1, 0, 0, a_4)$, where $a_1, a_4 \in \mathbb{F}^*_{2^r}$. Then we have $DP = PD'$, where $D' = diag(0, a_4, a_1, 0)$. Thus, $D + D' = diag(a_1, a_4, a_1, a_4)$, which is nonsingular. Hence, if D_2 has $t = \lceil \frac{n}{2} \rceil$ nonzero elements, then Lemma 5.4 will be modified as follows:*

$$M^r \leqq P^r + P^{r-1}D + P^{r-2}D + \ldots + PD + D_{\{t\}=0}$$

for $r \geq 2$, where D denotes some nonsingular diagonal matrix.

Theorem 5.2. *For an n-MDS DLS matrix $DLS(\rho; D_1, D_2)$ of order n, D_2 must have at least $\lceil \frac{n}{2} \rceil$ nonzero elements and ρ will be an n-cycle.*

Proof. Let $M = DLS(\rho; D_1, D_2)$ and D_2 has at most $t = \lceil \frac{n}{2} \rceil - 1$ nonzero elements. We have $M \leqq P + D_2$, where P is the permutation matrix corresponding to ρ. From Lemma 5.4, we have

$$M^n \leqq (P + D_2)^n \leqq P^n + P^{n-1}D + P^{n-2}D + \ldots + PD_{i=0} + D_{\{t\}=0}.$$

Therefore, we obtain

$$\begin{aligned} |M^n| &\leq |P^n + P^{n-1}D + P^{n-2}D + \ldots + PD_{i=0} + D_{\{t\}=0}| \\ &\leq |P^n| + |P^{n-1}D| + |P^{n-2}D| + \ldots + |PD_{i=0}| + |D_{\{t\}=0}|. \end{aligned}$$

Case 1: ρ is an n-cycle of S_n.

Then we have $P^n = I$, where I is the identity matrix of order n. Consequently, we have

$$\begin{aligned} |M^n| &\leq |I + P^{n-1}D + P^{n-2}D + \ldots + PD_{i=0} + D_{\{t\}=0}| \\ &\leq |I + D_{\{t\}=0}| + |P^{n-1}D| + |P^{n-2}D| + \ldots + |PD_{i=0}|. \end{aligned}$$

Note that $|I + D_{\{t\}=0}| = n$ and $|PD_{i=0}| = n - 1$. Hence,

$$\begin{aligned} |M^n| &\leq \underbrace{n + n + \ldots + n}_{n-1 \text{ times}} + (n-1) \\ &= (n-1)n + (n-1) = n^2 - 1 < n^2. \end{aligned}$$

Now, if D_2 has $\lceil \frac{n}{2} \rceil$ nonzero elements, then by Remark 5.5, we have

$$\begin{aligned} |M^n| &\leq |I + P^{n-1}D + P^{n-2}D + \ldots + PD + D_{\{t\}=0}| \\ &\leq |I + D_{\{t\}=0}| + |P^{n-1}D| + |P^{n-2}D| + \ldots + |PD| \\ &= n \cdot n = n^2. \end{aligned}$$

Thus, if D_2 has $\lceil \frac{n}{2} \rceil$ nonzero elements, then $DLS(\rho; D_1, D_2)$ of order n can potentially be n-MDS.

Case 2: ρ is not an n-cycle.

Then, from Corollary 5.1, we know that M cannot be k-MDS for $k \leq n$, even if D_2 has all n nonzero elements in its diagonal.

Therefore, $DLS(\rho; D_1, D_2)$ of order n can be n-MDS only when D_2 has at least $\lceil \frac{n}{2} \rceil$ nonzero elements and ρ is an n-cycle. □

5.2.1 Equivalence classes of DLS matrices to construct recursive MDS matrices

If the DLS matrix $DLS(\rho; D_1, D_2)$ of order n has fixed XOR $\mathcal{K} = l$, the diagonal of D_2 has l nonzero elements. Therefore, there are ${}^nC_l$ possible arrangements for distributing these l nonzero elements along the diagonal of D_2.

Also, in a DLS matrix $DLS(\rho; D_1, D_2)$, the permutation $\rho = [i_1, i_2, \ldots, i_n]$ must satisfy $i_k \neq k$ for $k = 1, 2, \ldots, n$. In other words, ρ represents a derangement of a set of n elements. Therefore, there are $D(n)$[1] possible choices for ρ in a DLS matrix, where $D(n)$ denotes the number of derangements of a set of n elements [RT09].

As a result, the search space for finding a recursive MDS matrix from the DLS matrices over the field $\mathbb{F}_{2^r}$ is given by $D(n) \cdot {}^nC_l \cdot (2^r)^{(n+l)}$. For example, the search space for finding a 6-MDS matrix from a DLS matrix of order 6, with $\mathcal{K} = 3$, over the field $\mathbb{F}_{2^4}$ is $265 \cdot 20 \cdot 2^{36} \approx 2^{48}$.

However, we have reduced the search space drastically by defining some equivalence classes of DLS matrices to construct recursive MDS matrices. Finally, we show that the existence of n-MDS DLS matrices over $\mathbb{F}_{2^r}$ with $\mathcal{K} = \left\lceil \frac{n}{2} \right\rceil$ implies the existence of n-MDS sparse DSI matrices over $\mathbb{F}_{2^r}$, and vice versa.

Theorem 5.3. *For $a_1, a_2, \ldots, a_n, a'_1, a'_2, \ldots, a'_n \in \mathbb{F}_{2^r}^*$ let $a = \prod_{i=1}^{n} a_i = \prod_{i=1}^{n} a'_i$ for some $a \in \mathbb{F}_{2^r}^*$. Then for any diagonal matrix D_2 over $\mathbb{F}_{2^r}$, the DLS matrix $M = DLS(\rho; D_1, D_2)$ of order n is n-MDS if and only if $M' = DLS(\rho; D'_1, D_2)$ is n-MDS, where $D_1 = diag(a_1, a_2, \ldots, a_n)$ and $D'_1 = diag(a'_1, a'_2, \ldots, a'_n)$.*

Proof. Suppose $\rho = [i_1, i_2, i_3, \ldots, i_n]$ and P is the permutation matrix corresponding to ρ. Now for any nonsingular diagonal matrix $D = diag(d_1, d_2, \ldots, d_n)$, we have

$$DMD^{-1} = D(PD_1 + D_2)D^{-1} = DPD_1D^{-1} + D_2.$$

Now by Lemma 2.10, we have $DP = PD'$ where $D' = diag(d_{i_1}, d_{i_2}, \ldots, d_{i_n})$. Thus, we have

$$DMD^{-1} = P(D'D_1D^{-1}) + D_2.$$

If $D'D_1D^{-1} = D'_1$, then we have

$$DMD^{-1} = PD'_1 + D_2 = M'. \tag{5.3}$$

[1]The formula for D_n is given by $D_n = (n-1)[D_{n-1} + D_{n-2}]$ with initial conditions $D_1 = 1$ and $D_0 = 0$. For example, the values of $D(n)$ are $1, 2, 9, 44$, and 265 for $n = 2, 3, 4, 5$, and 6, respectively.

Now if $D'D_1D^{-1} = D'_1$, we have $D'D_1 = D'_1D$. Therefore, we have

$$\left.\begin{array}{ll} a_1d_{i_1} & = a'_1d_1 \\ a_2d_{i_2} & = a'_2d_2 \\ & \vdots \\ a_nd_{i_n} & = a'_nd_n \end{array}\right\} \implies \left\{\begin{array}{ll} d_1 & = a_1d_{i_1}(a'_1)^{-1} \\ d_2 & = a_2d_{i_2}(a'_2)^{-1} \\ & \vdots \\ d_n & = a_nd_{i_n}(a'_n)^{-1}. \end{array}\right. \tag{5.4}$$

From Corollary 5.1, we know that a DLS matrix of order n can be n-MDS only when ρ is n-cycle. Thus, from above, we have

$$a_1a_2\dots a_n = a'_1a'_2\dots a'_n = a.$$

Now choosing $d_1 = 1$, from Equation 5.4, we get the values of other d_j's in terms of a_i's and a'_i's for $j = 2, 3, \dots, n$. Also, from Equation 5.3, we can say that M is n-MDS if and only if M' is n-MDS. □

Corollary 5.2. *For $a_1, a_2, \dots, a_n \in \mathbb{F}_{2^r}^*$ let $a = \prod_{i=1}^{n} a_i$ for some $a \in \mathbb{F}_{2^r}^*$. Then for any diagonal matrix D_2 over $\mathbb{F}_{2^r}$, the DLS matrix $M = DLS(\rho; D_1, D_2)$ of order n is n-MDS if and only if $M' = DLS(\rho; D'_1, D_2)$ is n-MDS, where $D_1 = diag(a_1, a_2, \dots, a_n)$ and $D'_1 = diag(a, 1, 1, \dots, 1)$.*

Remark 5.6. *For any $c \in \mathbb{F}_{2^r}^*$, M is n-MDS implies cM is also n-MDS. Thus, if ρ is an n-cycle permutation, $M = DLS(\rho; D_1, D_2)$ is diagonal similar to $M' = DLS(\rho; D'_1, D'_2)$, where $D_1 = diag(a_1, a_2, \dots, a_n)$, $D'_1 = diag(c^n a, 1, 1, \dots, 1)$, $D'_2 = c \cdot D_2$ and $a = \prod_{i=1}^{n} a_i$. We know that $x \to x^{2^l}$ is an isomorphism over $\mathbb{F}_{2^r}$. So when $n = 2^l$, there exist an element $c = a^{-1/n} \in \mathbb{F}_{2^r}^*$. Hence, when $n = 2^l$, we can say that M is diagonal similar to $M'' = DLS(\rho; D''_1, D''_2)$, where $D''_1 = diag(1, 1, 1, \dots, 1)$ and D''_2 is some diagonal matrix. Therefore, M is n-MDS if and only if M'' is also n-MDS.*

In [KSV19, Theorem 8], authors proved the same result as Theorem 5.3 and Corollary 5.2 for a fixed permutation $\rho = [2, 3, 4, \dots, n-1, n, 1]$. However, we have seen that the result holds for any n-cycle permutation.

Lemma 5.5. *Let $M_1 = DLS(\rho_1; D_1, D_2)$ be a DLS matrix of order n and $\rho_2 \in S_n$ is conjugate with ρ_1, then M_1 is n-MDS if and only if $M_2 = DLS(\rho_2; D'_1, D'_2)$ is n-MDS, where D'_1 and D'_2 are some diagonal matrices.*

Proof. Since ρ_1 and ρ_2 are conjugate, we have $\sigma\rho_1\sigma^{-1} = \rho_2$, for some $\sigma \in S_n$. Let P_1, P_2 and P be the permutation matrices related to ρ_1, ρ_2 and σ respectively. Then

we have

$$PM_1P^{-1} = P(P_1D_1 + D_2)P^{-1} = PP_1D_1P^{-1} + PD_2P^{-1}$$
$$= PP_1P^{-1}D_1^{'} + PP^{-1}D_2^{'},$$

where $D_1P^{-1} = P^{-1}D_1^{'}$ and $D_2P^{-1} = P^{-1}D_2^{'}$ for some diagonal matrices $D_1^{'}$ and $D_2^{'}$. Thus, we have $PM_1P^{-1} = P_2D_1^{'} + D_2^{'} = M_2$. Since $PM_1P^{-1} = M_2$, from Corollary 2.9 we can say that M_1 is n-MDS if and only if M_2 is n-MDS. □

Remark 5.7. *We know that a DLS matrix $DLS(\rho_1; D_1, D_2)$ can be n-MDS only when ρ is an n-cycle. Also, the n-cycles in $\mathcal{S}_n$ are conjugate to each other. Thus, for finding the n-MDS DLS matrices, we need to check only for the DLS matrices related to a particular n-cycle ρ.*

Now consider $\mathbb{D}(n, \mathbb{F}_{2^r})$ to be the set of all DLS matrices $DLS(\rho; D_1, D_2)$ of order n, with fixed XOR of k, over the field $\mathbb{F}_{2^r}$ and define

$$\mathbb{D}^{'}(n, \mathbb{F}_{2^r}) = \{B \in \mathbb{D}(n, \mathbb{F}_{2^r}) : \ B = P^{'}D_1^{'} + D_2^{'}\},$$

where $P^{'}$ is the permutation matrix related to the n length cycle $[2, 3, 4, \ldots, n-1, n, 1]$[2], $D_1^{'} = diag(a, 1, 1, \ldots, 1)$ and $D_2^{'}$ is a diagonal matrix containing k nonzero elements.

5.2.2 Equivalence of DLS matrices with sparse DSI matrices

In this section, we establish that the existence of n-MDS DLS matrices with $\mathcal{K} = \lceil \frac{n}{2} \rceil$ over $\mathbb{F}_{2^r}$ is equivalent to the existence of n-MDS sparse DSI matrices over $\mathbb{F}_{2^r}$.

Based on Corollary 5.2 and Remark 5.7, we can conclude that searching for n-MDS matrices within the set $\mathbb{D}(n, \mathbb{F}_{2^r})$, it suffices to focus on searching n-MDS matrices within the set $\mathbb{D}'(n, \mathbb{F}_{2^r})$. For the fixed XOR k, D_2 has ${}^nC_k$ many choices of arrangements of the k nonzero elements. However, from Theorem 4.4, we know that if $\rho = [2, 3, 4, \ldots, n-1, n, 1]$ and D_2 has any two consecutive zero entries [3], $B = PD_1 + D_2$ must contain a zero element when raised to power n i.e. B cannot be n-MDS, where P is the permutation matrix of order n related to the permutation ρ and D_1 is a nonsingular matrix. For the n-MDS DLS matrix $DLS(\rho; D_1, D_2)$ with $\rho = [2, 3, 4, \ldots, n-1, n, 1]$ and $\mathcal{K} = \lceil \frac{n}{2} \rceil$, there are n and 2 eligible arrangements

[2]By Remark 5.7, any n length cycle can be chosen for the set $\mathbb{D}^{'}(n, \mathbb{F}_{2^r})$.
[3]Note that here the first and n-th diagonal elements are also considered as consecutive entries.

Table 5.2: n-MDS DLS matrix of order n over the field $\mathbb{F}_{2^r}$ with $\mathcal{K} = \lceil \frac{n}{2} \rceil$ ("DNE" stands for does not exist).

Order n	**over $\mathbb{F}_{2^4}$**	**over $\mathbb{F}_{2^5}$**	**over $\mathbb{F}_{2^6}$**	**over $\mathbb{F}_{2^7}$**	**over $\mathbb{F}_{2^8}$**	**over $\mathbb{F}_{2^9}$**
4	Exists	Exists	Exists	Exists	Exists	Exists
5	DNE	Exists	Exists	Exists	Exists	Exists
6	DNE	DNE	DNE	Exists	Exists	Exists
7	DNE	DNE	DNE	DNE	Exists	Exists
8	DNE	DNE	DNE	DNE	DNE	Exists

of the nonzero elements in D_2 when n is odd and even respectively. However, we show that for all such eligible arrangements, $DLS(\rho; D_1, D_2)$ is permutation similar to some sparse DSI matrix.

Consider $\mathbb{D}''$ be the set of all DLS matrices $DLS(\rho; D_1, D_2)$ with $\rho = [2, 3, 4, \ldots, n-1, n, 1]$, $\mathcal{K} = \lceil \frac{n}{2} \rceil$, and the eligible arrangements of nonzero elements in D_2. It can be observed that any DLS matrix $B \in \mathbb{D}''$ are permutation similar to some sparse DSI matrix of order n. More specifically, we have

$$Q \cdot DLS(\rho; D_1, D_2) \cdot Q^{-1} = D_s,$$

where Q is the permutation matrix of order n related to the permutation $\sigma = \rho^k$ for some $k = 1, 2, \ldots, n$ and D_s denotes some sparse DSI matrix of order n.

Therefore, any DLS matrices $B \in \mathbb{D}''$ is n-MDS over $\mathbb{F}_{2^r}$ implies that there is an n-MDS sparse DSI matrix over $\mathbb{F}_{2^r}$. Hence, by Remark 5.7 and Table 4 of [KSV19], we have the results for the existence of n-MDS DLS matrices over $\mathbb{F}_{2^r}$ for $n = 4, 5, 6, 7, 8$ listed in Table 5.2.

Non-existence of n-MDS DLS matrices over $\mathbb{F}_{2^4}$ for $\lceil \frac{n}{2} \rceil < \mathcal{K} \leq n-1$: Companion matrices of order n have $\mathcal{K} = n-1$, whereas DLS matrices can be n-MDS if $\mathcal{K} = \lceil \frac{n}{2} \rceil$. In [TTKS18], the authors have provided some examples of efficient companion matrices that are n-MDS over the field $\mathbb{F}_{2^4}$ for $n = 5, 6, 7$, and 8. But from Table 5.2, we can see that there are no n-MDS DLS matrices with $\mathcal{K} = \lceil \frac{n}{2} \rceil$ over the field $\mathbb{F}_{2^4}$ for $n = 5, 6, 7$, and 8. Next, we increase the value of $\mathcal{K}$ from $\lceil \frac{n}{2} \rceil$ to $n-1$, to check whether there are n-MDS DLS matrices over the field $\mathbb{F}_{2^4}$. For this, we reduce the search space using Corollary 5.2 and Remark 5.7 and then perform an exhaustive search in the restricted domain. We observe that there are no n-MDS DLS matrices over the field $\mathbb{F}_{2^4}$ for $n = 5, 6, 7$, and 8 with $\mathcal{K} = \lceil \frac{n}{2} \rceil, \lceil \frac{n}{2} \rceil + 1, \ldots, n-1$.

5.3 Construction of Recursive MDS Matrices from GDLS Matrices

In this section, we extend the Definition 5.1 to propose a class of sparse matrices called as GDLS matrices. Then we propose some lightweight recursive MDS matrices of orders $4, 5, 6$, and 7, using GDLS matrices.

Definition 5.2. *(GDLS matrix) Consider two permutations* $\rho_1 = [i_1,\ i_2,\ i_3,\ \ldots\ , i_n]$ *and* $\rho_2 = [j_1,\ \ j_2,\ \ j_3,\ \ \ldots\ ,\ j_n]$ *such that* $i_k \neq j_k$ *for* $k = 1, 2, \ldots, n$. *Let* D_1 *be a nonsingular diagonal matrix and* D_2 *be a diagonal matrix (may be singular). Then we will call the matrix*

$$B = P_1 D_1 + P_2 D_2$$

as the generalized DLS (GDLS) matrix, where P_1 *and* P_2 *are the permutation matrices of order* n *related to the permutation* ρ_1 *and* ρ_2 *respectively. We will denote these matrices as* $GDLS(\rho_1, \rho_2; D_1, D_2)$.

Remark 5.8. *Note that GDLS matrix is row permuted matrix of DLS matrix. More specifically, we have*

$$GDLS(\rho_1, \rho_2; D_1, D_2) = P_2(P_2^{-1} P_1 D_1 + D_2) = P_2 \cdot DLS(\rho; D_1, D_2)$$

where $P = P_2^{-1} P_1$ *is the permutation matrix of order* n *related to the permutation* $\rho = \rho_2^{-1}\rho_1$. *Thus,* $GDLS(\rho_1, \rho_2; D_1, D_2)$ *is row permuted* $DLS(\rho_2^{-1}\rho_1; D_1, D_2)$.

Remark 5.9. *However, a* $GDLS(\rho_1, \rho_2; D_1, D_2)$ *matrix of order* n *is* n*-MDS does not imply that* $DLS(\rho_2^{-1}\rho_1; D_1, D_2)$ *is* n*-MDS. For* $\rho_1 = [2, 3, 4, 1], \rho_2 = [3, 2, 1, 4]$ *and* $D_1 = diag(1, 1, 1, \alpha^2), D_2 = diag(\alpha^{-1}, 0, \alpha^{-1}, 0)$, *we have the* 4*-MDS GDLS matrix*

$$\begin{aligned} M &= GDLS(\rho_1, \rho_2; D_1, D_2) = P_1 D_1 + P_2 D_2 \\ &= \begin{bmatrix} 0 & 0 & \alpha^{-1} & \alpha^2 \\ 1 & 0 & 0 & 0 \\ \alpha^{-1} & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \alpha^{-1} & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & \alpha^{-1} & \alpha^2 \\ 0 & 0 & 1 & 0 \end{bmatrix} \\ &= P_2 DLS(\rho_2^{-1}\rho_1; D_1, D_2), \end{aligned}$$

where $\rho_2^{-1}\rho_1 = [2, 1, 4, 3]$, α *is a primitive element of* $\mathbb{F}_{2^4}$ *with* $\alpha^4 + \alpha + 1 = 0$. *But the DLS matrix* $DLS(\rho_2^{-1}\rho_1; D_1, D_2)$ *is not* 4*-MDS.*

From the definition of GDLS matrices, it can be observed that the size of the set of all GDLS matrices with $\mathcal{K} = l$ over the field $\mathbb{F}_{2^r}$ is $n! \cdot D(n) \cdot {}^nC_l \cdot (2^r)^{(n+l)}$, where $D(n)$ represents the number of derangements for n distinct objects. This size is extremely large, making an exhaustive search impractical for obtaining a n-MDS matrix of order $n \geq 5$ from the GDLS matrices.

To minimize the search space, in most cases, we arbitrarily select ρ_1 as the n-cycle $[n,\ 1,\ 2,\ \ldots,\ n-1]$. However, it is important to note that there is no inherent advantage in choosing $\rho_1 = [n, 1, 2, \ldots, n-1]$ for obtaining a recursive MDS matrix. If we change $\rho_1 = [n, 1, 2, \ldots, n-1]$ to any permutation from S_n, there is still a possibility of obtaining a recursive MDS matrix.

Also to find lightweight recursive MDS matrices, we looked through the GDLS matrices of order n whose entries are from the set $\{1, \alpha, \alpha^{-1}, \alpha^2, \alpha^{-2}, \alpha^3, \alpha^{-3}\}$, where α is a primitive element of the field $\mathbb{F}_{2^r}$. First, we start with $\mathcal{K} = \left\lceil \frac{n}{2} \right\rceil$ and if we do not find any n-MDS GDLS matrix, then we increase the value of $\mathcal{K}$. Even with the set $\{1, \alpha, \alpha^{-1}, \alpha^2, \alpha^{-2}, \alpha^3, \alpha^{-3}\}$, the search space for finding n-MDS matrices of order $n \geq 5$ is large. Hence, we perform a random search to obtain n-MDS matrices of order $n = 5, 6, 7, 8$. But, the proposed 4-MDS matrix of order 4 is found by exhaustive search. Although we could not obtain n-MDS matrices from $\mathbb{F}_{2^4}$ for $n = 6, 7, 8$, and 8-MDS matrix over $\mathbb{F}_{2^8}$, there is still hope of getting such n-MDS matrices.

Note that the implementation costs of the matrices presented in this section over a field are calculated by referring to the s-XOR count value of the corresponding field elements as provided in table of [TTKS18, App. B].

5.3.1 Construction of 4×4 Recursive MDS Matrices

In this section, we propose a GDLS matrix of order 4 that yields a recursive MDS matrix. The proposed GDLS matrix is constructed by the permutations $\rho_1 = [4, 3, 1, 2]$, $\rho_2 = [2, 1, 4, 3]$ and with the value of $\mathcal{K} = 2$.

For $\rho_1 = [4, 3, 1, 2], \rho_2 = [2, 1, 4, 3]$ and diagonal matrices $D_1 = diag(\alpha, 1, 1, 1)$, $D_2 = diag(1, \alpha, 0, 0)$, we have the GDLS matrix of order 4 over $\mathbb{F}_{2^4}$ as follows

$$B = GDLS(\rho_1, \rho_2; D_1, D_2) = \begin{bmatrix} 0 & \alpha & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ \alpha & 0 & 0 & 0 \end{bmatrix},$$

where α a primitive element of $\mathbb{F}_{2^4}$ with $\alpha^4+\alpha+1=0$. The matrix B is a 4-MDS matrix with a XOR count of $(1+1)+2\cdot 4=10$.

Remark 5.10. *If we consider α to be a root of $x^8+x^7+x^6+x+1$ which is the constructing polynomial of $\mathbb{F}_{2^8}$, then B is 4-MDS over the field $\mathbb{F}_{2^8}$ and its XOR count will be $(3+3)+2\cdot 8=22$.*

Remark 5.11. *Due to the absence of trinomial irreducible polynomial of degree 8 over $\mathbb{F}_2$, elements with XOR count 1 in $\mathbb{F}_{2^8}$ are not possible (see [BKL16, Theorem 2]). But over rings, we can have elements with a XOR count of 1.*

Example 5.2. *Consider the GDLS matrix $B_{4,8}=GDLS(\rho_1,\rho_2;D_1,D_2)$ over $\mathbb{F}_2[L]$, where $\rho_1=[4,3,1,2],\rho_2=[2,1,4,3]$, $D_1=diag(L,1,1,1)$, $D_2=diag(1,L,0,0)$. Then it is easy to verify that $B_{4,8}^4$ is MDS over $\mathbb{F}_2[L]$. The set of minors of $B_{4,8}^4$ are*

$$\{1, L, L^2, L^2+1, L^2+L, L^3, L^3+1, L^3+L, L^3+L^2, L^3+L^2+L, L^4, L^4+L^2, L^4+L^3+1,$$
$$L^4+L^3+L, L^4+L^3+L^2, L^4+L^3+L^2+1, L^5, L^5+L^2, L^5+L^3, L^5+L^3+L, L^5+L^4,$$
$$L^5+L^4+L, L^5+L^4+L^2, L^5+L^4+L^3+L, L^6, L^6+L^4+L^2, L^6+L^5, L^7, L^8+L^3\}$$

whose factors are

$$\{1, L, L^2, L+1, L^2+L+1, L^3+L+1, L^3+L^2+1, L^4+L^3+1, L^4+L^3+L^2+L+1\}. \quad (5.5)$$

Now, consider the binary matrix $C_{4,8}=[[2],[3],[4],[5],[6],[7],[8],[1,3]]$ which is the companion matrix of x^8+x^2+1 over $\mathbb{F}_2$. Then using $L=C_{4,8}$, the given elements in 5.5 are nonsingular matrices over $\mathbb{F}_2$. In addition, the implementation cost of $C_{4,8}$ is 1 XOR. Hence, $B_{4,8}$ is 4-MDS over $GL(8,\mathbb{F}_2)$ and the implementation cost of $B_{4,8}$ is $(1+1)+2\cdot 8=18$ XORs.

5.3.2 Construction of 5×5 Recursive MDS Matrices

In this section, we propose a GDLS matrix of order 5 that yields a recursive MDS matrix over $\mathbb{F}_{2^8}$. This GDLS matrix exhibits the minimum XOR count when compared to the existing recursive MDS matrices of order 5 over $\mathbb{F}_{2^8}$. Also, we provide an example of a 5-MDS GDLS matrix over $\mathbb{F}_{2^4}$.

For $\rho_1=[3,4,5,1,2],\rho_2=[5,3,1,2,4]$ and diagonal matrices $D_1=diag(1,1,\alpha^{-1},1,1)$, $D_2=diag(\alpha,0,1,0,1)$, we have the GDLS matrix of order 5 over $\mathbb{F}_{2^8}$ as follows

$$B = GDLS(\rho_1, \rho_2; D_1, D_2) = \begin{bmatrix} 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ \alpha & 0 & \alpha^{-1} & 0 & 0 \end{bmatrix},$$

where α is a primitive element of $\mathbb{F}_{2^8}$ with $\alpha^8 + \alpha^7 + \alpha^6 + \alpha + 1 = 0$.

It can be observed that the matrix B is a 5-MDS matrix with a XOR count of $(3 + 3) + 3 \cdot 8 = 30$, outperforming the best known results for a 5-MDS matrix over $\mathbb{F}_{2^8}$.

It has been observed that there does not exist a 5-MDS DLS matrix of order 5 over the field $\mathbb{F}_{2^4}$ with $\mathcal{K} = 3$ and $\mathcal{K} = 4$. However, we have 5-MDS GDLS matrix of order 5 over the field $\mathbb{F}_{2^4}$ with $\mathcal{K} = 4$. For example for $\rho_1 = [4, 2, 3, 5, 1]$, $\rho_2 = [2, 4, 1, 3, 5]$ and diagonal matrices $D_1 = diag(\alpha, \alpha^3, 1, \alpha^2, \alpha^3)$, $D_2 = diag(1, 1, \alpha^{-1}, 1, 0)$, we have the GDLS matrix of order 5 over $\mathbb{F}_{2^4}$ as follows

$$B = \begin{bmatrix} 0 & 0 & \alpha^{-1} & 0 & \alpha^3 \\ 1 & \alpha^3 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ \alpha & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \alpha^2 & 0 \end{bmatrix},$$

where α is a primitive element of $\mathbb{F}_{2^4}$ with $\alpha^4 + \alpha + 1 = 0$. It can be verified that B is 5-MDS over $\mathbb{F}_{2^4}$ with XOR count of $(1 + 1 + 2 + 3 + 3) + 4 \cdot 4 = 26$.

Remark 5.12. *Consider the GDLS matrix $B_{5,8} = GDLS(\rho_1, \rho_2; D_1, D_2)$ over $\mathbb{F}_2[L]$, where $\rho_1 = [3, 4, 5, 1, 2], \rho_2 = [5, 3, 1, 2, 4]$, $D_1 = diag(1, 1, L^{-1}, 1, 1)$ and $D_2 = diag(L, 0, 1, 0, 1)$. Then it can be verified that $B_{5,8}$ is 5-MDS over $\mathbb{F}_2[L]$. Let $\mathbb{S}_5$ be the set of factors of the minors of $B_{5,8}^5$. It is easy to check that the polynomial $L^8 + L^7 + L^2 + L + 1 \notin \mathbb{S}_5$.*

Now, consider the binary matrix $C_{5,8} = [[8], [1, 2], [2, 8], [3], [4], [5], [6], [7]]$ whose minimal polynomial is $x^8 + x^7 + x^2 + x + 1$. Then using $L = C_{5,8}$, the elements in $\mathbb{S}_5$ are nonsingular matrices over $\mathbb{F}_2$. In addition, the implementation cost of $C_{5,8}$ is 2 XORs. Also, $C_{5,8}^{-1}$ can be implemented with 2 XORs. Hence, $B_{5,8}$ is 5-MDS over $GL(8, \mathbb{F}_2)$ and the implementation cost of $B_{5,8}$ is $(2 + 2) + 3 \cdot 8 = 28$ XORs.

5.3.3 Construction of 6×6 Recursive MDS Matrices

In this section, we propose a GDLS matrix of order 6 that yields a recursive MDS matrix over $\mathbb{F}_{2^8}$.

For $\rho_1 = [6, 1, 2, 3, 4, 5], \rho_2 = [5, 6, 1, 2, 3, 4]$ and diagonal matrices $D_1 = diag(1, 1, \alpha, 1, 1, 1), D_2 = diag(1, 0, 1, 0, \alpha^2, 0)$, we have the GDLS matrix of order 6 over $\mathbb{F}_{2^8}$ as follows

$$B = GDLS(\rho_1, \rho_2; D_1, D_2) = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & \alpha & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & \alpha^2 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix},$$

where α is a primitive element of $\mathbb{F}_{2^8}$ with $\alpha^8 + \alpha^7 + \alpha^6 + \alpha + 1 = 0$.

It can be observed that the matrix B is a 6-MDS matrix with a XOR count of $(3+4)+3\cdot 8 = 31$, which corresponds to the best result for a 6-MDS matrix over $\mathbb{F}_{2^8}$.

Remark 5.13. *Consider the GDLS matrix $B_{6,8} = GDLS(\rho_1, \rho_2; D_1, D_2)$ over $\mathbb{F}_2[L]$, where $\rho_1 = [6, 1, 2, 3, 4, 5], \rho_2 = [5, 6, 1, 2, 3, 4]$ and $D_1 = diag(1, 1, L, 1, 1, 1)$, $D_2 = diag(1, 0, 1, 0, L^2, 0)$. Then it can be verified that $B_{6,8}$ is 6-MDS over $\mathbb{F}_2[L]$. Let $\mathbb{S}_6$ be the set of factors of the minors of $B_{6,8}^6$. It is easy to check that the polynomial $L^8 + L^7 + L^2 + L + 1 \notin \mathbb{S}_6$. Then using $L = C_{5,8}$ from Remark 5.12 the elements in $\mathbb{S}_6$ are nonsingular matrices over $\mathbb{F}_2$. The binary matrix $C_{5,8}^2$ can be implemented with 4 XORs. Hence, $B_{6,8}$ is 6-MDS over $GL(8, \mathbb{F}_2)$ and the implementation cost of $B_{6,8}$ is $(2+4)+3\cdot 8 = 30$ XORs.*

Following that, we looked for a 6-MDS GDLS matrix over $\mathbb{F}_{2^4}$, but we could not find a 6-MDS matrix of order 6 in $\mathbb{F}_{2^4}$.

5.3.4 Construction of 7×7 Recursive MDS Matrices

Here, we propose a GDLS matrix of order 7 that yields a recursive MDS matrix over $\mathbb{F}_{2^8}$.

For $\rho_1 = [7, 1, 2, 3, 4, 5, 6], \rho_2 = [3, 2, 5, 4, 7, 6, 1]$ and diagonal matrices $D_1 = diag(1, \alpha^{-2}, \alpha^{-2}, 1, \alpha^3, 1, 1), D_2 = diag(1, 0, 1, 0, 1, 0, 1)$, we have the GDLS matrix of order 7 over $\mathbb{F}_{2^8}$ as follows

$$B = GDLS(\rho_1, \rho_2; D_1, D_2) = \begin{bmatrix} 0 & \alpha^{-2} & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & \alpha^{-2} & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \alpha^3 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix},$$

where α is a primitive element of $\mathbb{F}_{2^8}$ with $\alpha^8 + \alpha^7 + \alpha^6 + \alpha + 1 = 0$.

It can be observed that the matrix B is a 7-MDS matrix with a XOR count of $(5 + 4 + 4) + 4 \cdot 8 = 45$, outperforming the best known results for a 7-MDS matrix over $\mathbb{F}_{2^8}$.

Afterwards, we attempted to find a 7-MDS GDLS matrix over $\mathbb{F}_{2^4}$, but we were unable to obtain a 7-MDS matrix of order 7 in $\mathbb{F}_{2^4}$.

As 4 and 8 are the most commonly used diffusion layer matrix sizes, we look for an 8-MDS GDLS matrix of order 8 over $\mathbb{F}_{2^4}$ and $\mathbb{F}_{2^8}$. However, our search did not yield a GDLS matrix of order 8 that corresponds to an 8-MDS matrix.

5.3.5 Importance of GDLS Matrices

In this section, we look at the importance of GDLS matrices for the construction of recursive MDS matrices.

1. The popular DSI matrix is a particular type of GDLS matrix. Additionally, the GFS matrix structure utilized in [WWW13] can be seen as a particular type of GDLS matrix. Specifically, for $\rho_1 = [n, 1, 2, 3, \ldots, n-1]$, $\rho_2 = [1, n, 3, 2, 5, 4, 7, \ldots, n-1, n-2]$, a nonsingular diagonal matrix $D_1 = diag(L_1, 1, L_3, 1, \ldots, L_{n-1}, 1)$ and $D_2 = diag(0, L_2, 0, L_4, 0, \ldots, 0, L_n)$ we have $GDLS(\rho_1, \rho_2; D_1, D_2)$ as the GFS matrix in [WWW13]. Furthermore, each GFS matrix structure used in [SM21] to construct MDS matrices can be viewed as a GDLS matrix.
2. The GDLS matrix structure is not limited to even orders, unlike the GFS matrix structure used in [SM21, WWW13]. The GDLS matrix structure is applicable to matrices of all orders, enabling improvements in several parameters that were not achievable with the GFS matrix structure.

3. From Table 5.2 and subsequent discussions, we observe that there are no n-MDS DLS matrices over $\mathbb{F}_{2^4}$ for $n = 5, 6, 7, 8$ with $\mathcal{K} = \lceil \frac{n}{2} \rceil, \lceil \frac{n}{2} \rceil + 1, \ldots, n-1$. However, we do have 5-MDS GDLS matrices over $\mathbb{F}_{2^4}$ with a fixed XOR of 4. Additionally, the diagonal or permutation similar matrix of a DLS matrix is again a DLS matrix. Hence, the structure of GDLS matrices is crucial for the construction of recursive MDS matrices.

4. From Theorem 4.1, we know that there are no n-MDS matrices of order n with fixed XOR of 1 for $n \geq 3$. Thus, the example of the 4-MDS GDLS matrix in Section 5.3.1 possesses the lowest possible fixed XOR. Given that the diffusion matrix of order 4 is commonly utilized in the diffusion layer, the 4-MDS GDLS matrix becomes an excellent choice for designing lightweight ciphers.

5. Using GDLS matrices, we provide some lightweight recursive MDS matrices of orders $5, 6$, and 7 over $\mathbb{F}_{2^8}$. The results match those of the best known lightweight recursive MDS matrices of order 6 and outperform those of orders 5 and 7.

5.4 Conclusion

This chapter comprehensively studies DLS matrices for constructing recursive MDS matrices. To address the impracticality of exhaustive searches, several theoretical results are presented that reduce the search space to a smaller domain. It is shown that for $n = 5, 6, 7, 8$ and $\mathcal{K} = \left\lceil \frac{n}{2} \right\rceil, \left\lceil \frac{n}{2} \right\rceil + 1, \ldots, n-1$, there are no n-MDS DLS matrices of order n over the field $\mathbb{F}_{2^4}$. Additionally, it is demonstrated that an n-MDS DLS matrix over $\mathbb{F}_{2^r}$ with $\mathcal{K} = \left\lceil \frac{n}{2} \right\rceil$ is a permutation similar to some n-MDS sparse DSI matrix. Moreover, the importance of GDLS matrices for constructing recursive MDS matrices is discussed, and efficient recursive MDS matrices are provided for various orders.

Our investigations in this chapter open up many possibilities for future work.

1. We see that 8-MDS DLS matrices do not exist over $\mathbb{F}_{2^8}$ with the lowest fixed XOR of 4. This leads to a potential future investigation to determine if there exists an 8-MDS DLS matrix of order 8 with a higher fixed XOR over $\mathbb{F}_{2^8}$.

2. Theoretical results for DLS matrices have been presented in order to narrow the search space for the finding n-MDS DLS matrices over $\mathbb{F}_{2^r}$. It could be a future research direction to provide theoretical results on GDLS matrices for a similar purpose. Since 4 and 8 are the most commonly used sizes for diffusion

layer matrices, we searched for an 8-MDS GDLS matrix of order 8 over $\mathbb{F}_{2^4}$ and $\mathbb{F}_{2^8}$, but our search did not yield any success. Therefore, it might be a potential future work to find or prove the non-existence of 8-MDS GDLS matrices of order 8 over $\mathbb{F}_{2^4}$ and $\mathbb{F}_{2^8}$.

3. By utilizing a composition of different GFS matrices, lightweight MDS matrices of even orders (4, 6, and 8) have been constructed by the authors in [SM21]. Since the GDLS matrix structure exists for all orders, it could be a potential area of future research to explore the use of GDLS matrices in these constructions and find lightweight MDS matrices of any order.

4. Many direct constructions of recursive MDS matrices from companion matrices are known. Thus, a potential research problem is to find a direct construction method for recursive MDS matrices from the DLS or GDLS matrices.

6

Near-MDS Matrices: A Comprehensive Study of Properties and Designs

Contents

6.1 Introduction

Due to their optimal branch number, MDS matrices are widely preferred for the construction of diffusion layers in block ciphers and hash functions. However, in lightweight cryptography, the balance between security and efficiency may not be optimal when using MDS matrices. Near-MDS (NMDS) matrices, on the other hand, have sub-optimal branch numbers, resulting in slower diffusion speed and a smaller minimum number of active Sboxes per round compared to ciphers that utilize MDS matrices.

Nevertheless, studies such as [ABI+18, BBI+15] have demonstrated that the incorporation of NMDS matrices, combined with a carefully selected permutation, can

enhance security against differential and linear cryptanalysis. This makes NMDS matrices a better choice for achieving a balance between security and efficiency in the diffusion layer of lightweight cryptographic primitives. Several recent lightweight block ciphers, such as PRIDE [ADK+14], Midori [BBI+15], MANTIS [BJK+16], FIDES [BBK+13], and PRINCE [BCG+12], have employed NMDS matrices in their diffusion layer. With the increasing importance of lightweight symmetric key primitives that prioritize low power consumption, energy efficiency, or low latency, the utilization of NMDS matrices in the construction of lightweight block ciphers has become more prevalent. However, the existing literature has limited studies on NMDS and recursive NMDS matrices. This serves as our motivation to present new results on NMDS matrices.

In this chapter, our focus is on studying NMDS matrices and exploring their construction in both recursive and nonrecursive settings. We present several theoretical results and analyze the hardware efficiency of NMDS matrix construction. Throughout the study, we make comparisons between NMDS matrices and MDS matrices whenever feasible.

Regarding the recursive approach, we investigate the DLS matrices and establish some theoretical results that aid in restricting the search space of the DLS matrices. Furthermore, we demonstrate that any sparse matrix of order $n \geq 4$ with fixed XOR value of 1 over a field of characteristic 2 cannot be an NMDS when raised to a power of $k \leq n$. We then employ the GDLS matrices to construct some lightweight recursive NMDS matrices of different orders that perform better than the existing matrices in terms of hardware cost or number of iterations.

For the nonrecursive construction of NMDS matrices, we examine various structures such as circulant and left-circulant matrices, as well as their generalizations such as Toeplitz and Hankel matrices. We also prove that Toeplitz matrices of order $n > 4$ cannot be both NMDS and involutory over a field of characteristic 2. Finally, we utilize GDLS matrices to construct some lightweight NMDS matrices that can be computed in a single clock cycle. The nonrecursive NMDS matrices proposed for orders 4, 5, 6, 7, and 8 can be implemented using 24, 50, 65, 96, and 108 XORs over $\mathbb{F}_{2^4}$, respectively.

Outline: The rest of this chapter is structured as follows: Section 6.2 delves into the discussion of DLS matrices for constructing recursive NMDS matrices. In Section 6.3, we present several lightweight recursive NMDS matrices of various orders by utilizing GDLS matrices. Section 6.4 explores the utilization of circulant, left-

circulant, Toeplitz, and Hankel matrices in the construction of nonrecursive NMDS matrices. Furthermore, in Section 6.5, we present lightweight nonrecursive NMDS matrices constructed from GDLS matrices. Finally, Section 6.6 concludes the chapter.

6.2 Construction of Recursive NMDS Matrices from DLS Matrices

The construction of recursive MDS matrices has received considerable attention in the literature, as demonstrated by various works [AF15, Ber13, GPP11, GPPR11, GPV17a, GPV17b, GPV19, KPSV21, TTKS18, WWW13]. However, there has been limited research on the construction of NMDS and recursive NMDS matrices. In this section, we focus on utilizing DLS matrices for constructing recursive NMDS matrices.

In a recent work [LW21], the authors presented some lightweight recursive NMDS matrices with the lowest fixed XOR value (i.e., $\mathcal{K} = 1$). However, these matrices require a large number of iterations, making them unsuitable for low-latency applications. In this chapter, we consider the case of checking whether B^k is NMDS or not for $k \leq n$.

From Corollary 2.3, we know that any matrix of order n cannot be NMDS if the number of nonzero entries is less than $n^2 - n$. We are using this fact for the proof of the following theorem.

Theorem 6.1. *Given a DLS matrix $M = DLS(\rho; D_1, D_2)$ of order $n \geq 2$, M^k is not NMDS for $k < n - 2$.*

Proof. We have $M = DLS(\rho; D_1, D_2) \leqq P + D_2$, where P is the permutation matrix corresponding to ρ. From Lemma 5.1, we have

$$\begin{aligned} |M^k| &\leq |P^k + P^{k-1}D + \cdots + PD + D_2^2| \\ &\leq |P^k D| + |P^{k-1}D| + \cdots + |PD| + |D_2^2|. \end{aligned} \tag{6.1}$$

Since power of a permutation matrix is again a permutation matrix, we have

$$|M^k| \leq \underbrace{|D| + |D| + \cdots + |D|}_{k \text{ times}} + |D_2^2| \leq kn + n. \tag{6.2}$$

Now for $k < n-2$, we have

$$|M^k| < (n-2)n + n = n^2 - n \implies |M^k| < n^2 - n.$$

Hence, M^k is not NMDS for $k < n-2$. □

Remark 6.1. *For comparison with recursive MDS matrices, it can be observed that a DLS matrix of order n requires a minimum power of $n-2$ to be an NMDS matrix and a minimum power of $n-1$ to be an MDS matrix.*

Remark 6.2. *From Theorem 6.1, we know that for a DLS matrix $M = DLS(\rho; D_1, D_2)$ of order $n \geq 2$, M^k is not an NMDS for $k < n-2$. However, there exist k-NMDS DLS matrix for $k = n-2$. For example, consider the DLS matrix $M = DLS(\rho; D_1, D_2)$ of order 4 with $\rho = [4, 1, 2, 3]$, $D_1 = diag(\alpha^2, \alpha^2, \alpha^2, \alpha^2)$ and $D_2 = diag(\alpha^2, 1, \alpha^2, 1)$, where α is a primitive element of $\mathbb{F}_{2^4}$ with $\alpha^4 + \alpha + 1 = 0$. It can be checked that the matrix*

$$\begin{aligned} M &= DLS(\rho; D_1, D_2) \\ &= \begin{bmatrix} \alpha^2 & \alpha^2 & 0 & 0 \\ 0 & 1 & \alpha^2 & 0 \\ 0 & 0 & \alpha^2 & \alpha^2 \\ \alpha^2 & 0 & 0 & 1 \end{bmatrix} \end{aligned}$$

is 2-NMDS.

Based on Lemma 5.2 and Remark 5.3, we can derive the following result regarding the influence of the permutation ρ in a DLS matrix $DLS(\rho; D_1, D_2)$ on the construction of recursive NMDS matrices.

Corollary 6.1. *For a DLS matrix $M = DLS(\rho; D_1, D_2)$ of order $n \geq 2$, if ρ is not an n-cycle, then M^k is not NMDS for $k \leq n$.*

So far, we have not considered the presence of zero entries in the diagonal of D_2. Now, let us examine the scenario where D_2 is singular.

Lemma 6.1. *In a DLS matrix $M = DLS(\rho_1; D_1, D_2)$ of order $n \geq 2$, if D_2 is singular, then M^k cannot be NMDS for $k \leq n-2$, even if ρ is n-cycle.*

Proof. If D_2 is singular, having at least one zero in the diagonal then from Equa-

tion 6.1 we have,

$$\begin{aligned} |M^k| &\leq |P^k D| + |P^{k-1} D| + \cdots + |PD| + |D_{i=0}| \\ &\leq \underbrace{n + n + \cdots + n}_{k \text{ times}} + (n-1) = kn + n - 1. \end{aligned} \tag{6.3}$$

Where $D_{i=0}$ be some diagonal matrix with a zero at the i-th diagonal position for some $i \in \{1, 2, \ldots, n\}$. Thus, for $k \leq n-2$, we have $|M^k| < n^2 - n$. Hence, M cannot be k-NMDS for $k \leq n-2$. □

Remark 6.3. *When comparing with recursive MDS matrices, it can be observed that for a DLS matrix $B = DLS(\rho; D_1, D_2)$ of order n, if ρ is not an n-cycle, then B^k is neither MDS nor NMDS for $k \leq n$. Additionally, if D_2 is singular, then B requires a minimum power of $n-1$ to be an NMDS matrix and a minimum power of n to be an MDS matrix.*

Remark 6.4. *Based on the Theorem 6.1, it is established that for a DLS matrix $M = DLS(\rho; D_1, D_2)$ of order $n \geq 2$, M^k is not an NMDS matrix when $k < n-2$. However, there exists a DLS matrix that is a k-NMDS matrix when $k = n-2$. For example, consider the DLS matrix $M = DLS(\rho; D_1, D_2)$ of order 4 with $\rho = [4, 1, 2, 3]$, $D_1 = diag(\alpha^2, \alpha^2, \alpha^2, \alpha^2)$ and $D_2 = diag(\alpha^2, 1, \alpha^2, 1)$, where α is a primitive element of $\mathbb{F}_{2^4}$ with $\alpha^4 + \alpha + 1 = 0$. It can be checked that the matrix*

$$\begin{aligned} M &= DLS(\rho; D_1, D_2) \\ &= \begin{bmatrix} \alpha^2 & \alpha^2 & 0 & 0 \\ 0 & 1 & \alpha^2 & 0 \\ 0 & 0 & \alpha^2 & \alpha^2 \\ \alpha^2 & 0 & 0 & 1 \end{bmatrix} \end{aligned}$$

is 2-NMDS.

Discussion: From Theorem 5.2, we know that for an n-MDS DLS matrix of order n, we must have $\mathcal{K} \geq \lceil \frac{n}{2} \rceil$. However, the minimum value of $\mathcal{K}$ may be less than $\lceil \frac{n}{2} \rceil$ for having at least $n^2 - n$ nonzero elements when it is raised to power $n-1$. For example, consider the DLS matrix $B = DLS(\rho; D_1, D_2)$ of order 5 over $\mathbb{F}_{2^4}/0\text{x}13$, where $\rho = [5, 1, 2, 3, 4]$, $D_1 = diag(\alpha^4, \alpha^4, 1, 1, 1)$, $D_2 = diag(\alpha, 0, 0, 1, 0)$ and $\alpha^4 + \alpha + 1 = 0$. Then B^4 have 21 nonzero elements. However, B^4 is not an NMDS matrix.

For NMDS matrices, we could not find the minimum value of $\mathcal{K}$ like we have for MDS matrices. In the next section, we will provide some theoretical results about NMDS matrices that will help us in determining the minimum value of $\mathcal{K}$ for DLS matrices of order n that are k-NMDS with $k = n - 1$ and $k = n$.

Remark 6.5. *From Theorem 4.1, we know that any matrix M of order n with $\mathcal{K} = 1$ can have at most $\frac{n(n+3)}{2} - 1$ nonzero elements when it raised to the power n. Hence, for $n \geq 5$, we have $|M^n| < n^2 - n$. Thus, for $n \geq 5$, any matrix of order n with $\mathcal{K} = 1$ cannot be n-NMDS.*

For $n = 4$, we have $\frac{n(n+3)}{2} - 1 > n^2 - n$. So it may seem that a matrix of order 4 with $\mathcal{K} = 1$ can be 4-NMDS. However, in the following theorem, we will see that to be 4-NMDS, a matrix of order 4 must have $\mathcal{K} = 2$.

Theorem 6.2. *There does not exist any 4-NMDS matrix of order 4 with $\mathcal{K} = 1$ over a field of characteristic 2.*

Proof. A matrix M of order n can never be recursive NMDS if its one row or column has all zero entries [1]. Also, if M contains n nonzero elements in such a way that no column or row has all zero entries, then M is of the form $M = PD$, where P represents a permutation matrix and D represents a diagonal matrix. Then by Lemma 2.10, any power of M is again of the form $P'D'$, for some permutation matrix P' and diagonal matrix D'. Hence, M cannot be recursive NMDS.

Let $\mathcal{S}$ be the set of all matrices M that contain $n+1$ nonzero elements with $\mathcal{K} = 1$ and in such a way that no column or row has all zero entries. Then each $M \in \mathcal{S}$ can be written as $M = PD + A$, where A has only one nonzero element. Let the nonzero element lies in the i-th row of A.

Now consider the permutation matrix P_1 obtained from the identity matrix by permuting the row i to row 1. Now

$$\begin{aligned} P_1 M P_1^{-1} &= P_1(PD + A)P_1^{-1} \\ &= P_1 P D P_1^{-1} + P_1 A P_1^{-1} \end{aligned}$$

By Lemma 2.10, we have $DP_1^{-1} = P_1^{-1}D'$ for some diagonal matrix D'. Thus, we

[1] The theorem states about the matrices of order $n = 4$ and the first part of the proof holds for any matrix of order n.

have

$$\begin{aligned} P_1 M P_1^{-1} &= P_1 P P_1^{-1} D' + P_1 A P_1^{-1} \\ &= QD' + A', \end{aligned}$$

where $Q = P_1 P P_1^{-1}$, $A' = P_1 A P_1^{-1}$ and A' has the nonzero element in its first row. Therefore, M is permutation similar to $QD' + A'$. Now let $\mathcal{S}' \subset \mathcal{S}$ be the set of all matrices with two nonzero elements in the first row.

Since from Fact 2.7, we know that a permutation similar to a recursive NMDS matrix is also a recursive NMDS matrix, we simply need to check from the set $\mathcal{S}'$ for finding all recursive NMDS matrices with $\mathcal{K} = 1$.

It can be checked that there are only six[2] matrix structures (See (6.4)) of order $n = 4$ from the set $\mathcal{S}'$ that can potentially be NMDS (i.e. number of nonzero elements > 12) when they are raised to power 4. However, all the six structures

$$\begin{bmatrix} * & * & 0 & 0 \\ 0 & 0 & * & 0 \\ 0 & 0 & 0 & * \\ * & 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} * & 0 & 0 & * \\ * & 0 & 0 & 0 \\ 0 & * & 0 & 0 \\ 0 & 0 & * & 0 \end{bmatrix}, \begin{bmatrix} * & * & 0 & 0 \\ 0 & 0 & 0 & * \\ * & 0 & 0 & 0 \\ 0 & 0 & * & 0 \end{bmatrix}, \begin{bmatrix} * & 0 & * & 0 \\ * & 0 & 0 & 0 \\ 0 & 0 & 0 & * \\ 0 & * & 0 & 0 \end{bmatrix}, \begin{bmatrix} * & 0 & 0 & * \\ 0 & 0 & * & 0 \\ * & 0 & 0 & 0 \\ 0 & * & 0 & 0 \end{bmatrix} \text{ and } \begin{bmatrix} * & 0 & * & 0 \\ 0 & 0 & 0 & * \\ 0 & * & 0 & 0 \\ * & 0 & 0 & 0 \end{bmatrix} \tag{6.4}$$

are also permutation similar. Now consider the first matrix structure and let

$$M = \begin{bmatrix} a & x_1 & 0 & 0 \\ 0 & 0 & x_2 & 0 \\ 0 & 0 & 0 & x_3 \\ x_4 & 0 & 0 & 0 \end{bmatrix},$$

where a, x_1, x_2, x_3 and x_4 are some nonzero elements in the field. Now consider the input vector of M as $[1, ax_1^{-1}, 0, 0]^T$. The resultant vector after each iteration is

$$\begin{bmatrix} 1 \\ ax_1^{-1} \\ 0 \\ 0 \end{bmatrix} \xrightarrow[\text{i=1}]{} \begin{bmatrix} 0 \\ 0 \\ 0 \\ x_4 \end{bmatrix} \xrightarrow[\text{i=2}]{} \begin{bmatrix} 0 \\ 0 \\ x_3x_4 \\ 0 \end{bmatrix} \xrightarrow[\text{i=3}]{} \begin{bmatrix} 0 \\ x_2x_3x_4 \\ 0 \\ 0 \end{bmatrix} \xrightarrow[\text{i=4}]{} \begin{bmatrix} x_1x_2x_3x_4 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

The sum of nonzero elements of input vector and output vector in each iteration is less than 4 i.e. branch number of $M < 4$. Therefore, M is not k-NMDS for $k \leq 4$. Hence, there does not exist any 4-NMDS matrix of order 4 with $\mathcal{K} = 1$ over a field of

[2]For $n = 4$, there are a total of 72 elements in $\mathcal{S}'$, and by running a computer search, we have observed that there are only 6 matrix structures that have at least 12 nonzero elements when raised to the power 4.

characteristic 2. □

From Lemma 4.2, we can easily check that for a matrix of order $n \geq 4$ with $\mathcal{K} = 1$, $|M^k| < n^2 - n$ for $k \leq n - 1$. Thus, by using Remark 6.5 and Theorem 6.2, we can conclude the following theorem.

Theorem 6.3. *For $n \geq 4$, there does not exist any k-NMDS matrix of order n with $\mathcal{K} = 1$ and $k \leq n$ over a field of characteristic 2.*

Remark 6.6. *For $n < 4$, there may exist a k-NMDS matrix of order n with $\mathcal{K} = 1$ and $k \leq n$. For example, the matrix*

$$B = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$$

is a 3-NMDS matrix.

Fact 6.1. *Over a field of characteristic 2, a DLS matrix of order n with $\mathcal{K} = 1$ cannot be k-NMDS for $n \geq 4$ and $k \leq n$.*

Now we will discuss some equivalence classes of DLS matrices for the construction of recursive NMDS matrices.

6.2.1 Equivalence classes of DLS matrices

If the DLS matrix $DLS(\rho; D_1, D_2)$ of order n has a fixed XOR of l, the diagonal of D_2 has l nonzero elements. Since there are ${}^nC_l$ arrangements for the l nonzero elements in the diagonal of D_2, the search space for finding a recursive NMDS matrix from the DLS matrices over the field $\mathbb{F}_{2^r}$ is $D(n) \cdot {}^nC_l \cdot (2^r)^{(n+l)}$, where $D(n)$ is the number of derangements for n different objects (see Section 5.2.1). However, we have drastically reduced the search space by defining some equivalence classes of DLS matrices.

From Fact 2.7 we know that diagonal similar of a NMDS matrix is again a NMDS matrix. Hence, we can apply the results of Theorem 5.3 and Corollary 5.2 to also the NMDS matrices.

Corollary 6.2. *Let $a = \prod_{i=1}^{n} a_i$ for some $a_1, a_2, \ldots, a_n \in \mathbb{F}_{2^r}^*$. Then for any diagonal matrix D_2 over $\mathbb{F}_{2^r}$, the DLS matrix $M = DLS(\rho; D_1, D_2)$ of order n is k-NMDS if and only if $M' = DLS(\rho; D_1', D_2)$ is k-NMDS, where $k \in \{n-1, n\}$, $D_1 = diag(a_1, a_2, \ldots, a_n)$ and $D_1' = diag(a, 1, 1, \ldots, 1)$.*

Remark 6.7. *For any $c \in \mathbb{F}_{2^r}^*$, M is k-NMDS implies cM is also k-NMDS. Thus, if ρ is an n-cycle permutation, $M = DLS(\rho; D_1, D_2)$ is diagonal similar to $M' = DLS(\rho; D_1', D_2')$, where $D_1 = diag(a_1, a_2, \ldots, a_n)$, $D_1' = diag(c^n a, 1, 1, \ldots, 1)$, $D_2' = c \cdot D_2$ and $a = \prod_{i=1}^{n} a_i$. We know that $x \to x^{2^l}$ is an isomorphism over $\mathbb{F}_{2^r}$. So when $n = 2^l$, there exist an element $c = a^{-1/n} \in \mathbb{F}_{2^r}^*$. Hence, when $n = 2^l$, we can say that M is diagonal similar to $M'' = DLS(\rho; D_1'', D_2'')$, where $D_1'' = diag(1, 1, 1, \ldots, 1)$ and D_2'' is some diagonal matrix. Therefore, for $k \in \{n-1, n\}$, M is k-NMDS if and only if M'' is also k-NMDS.*

We know that a permutation similar to an NMDS matrix is again an NMDS matrix. Thus, we can reduce the search space further by eliminating the permutation similar matrices from the search space. To accomplish this, we require the lemma provided below.

Lemma 6.2. *Let $M_1 = DLS(\rho_1; D_1, D_2)$ be a DLS matrix of order n and $\rho_2 \in S_n$ is conjugate with ρ_1, then M_1 is k-NMDS if and only if $M_2 = DLS(\rho_2; D_1', D_2')$ is k-NMDS, where D_1' and D_2' are some diagonal matrices.*

Proof. Since ρ_1 and ρ_2 are conjugate, we have $\sigma\rho_1\sigma^{-1} = \rho_2$, for some $\sigma \in S_n$. Let P_1, P_2 and P be the permutation matrices related to ρ_1, ρ_2 and σ respectively. Then we have

$$\begin{aligned} PM_1P^{-1} &= P(P_1D_1 + D_2)P^{-1} = PP_1D_1P^{-1} + PD_2P^{-1} \\ &= PP_1P^{-1}D_1' + PP^{-1}D_2', \end{aligned}$$

where $D_1P^{-1} = P^{-1}D_1'$ and $D_2P^{-1} = P^{-1}D_2'$ for some diagonal matrices D_1' and D_2'. Thus, we have $PM_1P^{-1} = P_2D_1' + D_2' = M_2$. Since $PM_1P^{-1} = M_2$, from Corollary 2.9 we can say that M_1 is k-NMDS if and only if M_2 is k-NMDS. □

Remark 6.8. *If D_2 is singular, a DLS matrix $M = DLS(\rho_1; D_1, D_2)$ of order n cannot be k-NMDS for $k \leq n-2$. Also, ρ must be an n-cycle for M to be k-NMDS with $k = n-1$ or $k = n$. In addition, the n-cycles in $\mathcal{S}_n$ are conjugate with each other. Therefore, to find the k-NMDS (with $k = n-1$ and $k = n$) DLS matrices, we need to check only for the DLS matrices associated with one fixed n-cycle ρ.*

Now consider $\mathbb{D}(n, \mathbb{F}_{2^r})$ to be the set of all DLS matrices $DLS(\rho; D_1, D_2)$ of order n, with $\mathcal{K} = \lceil \frac{n}{2} \rceil$, over the field $\mathbb{F}_{2^r}$ and define

$$\mathbb{D}'(n, \mathbb{F}_{2^r}) = \{B \in \mathbb{D}(n, \mathbb{F}_{2^r}) : \ B = P'D_1' + D_2'\},$$

where $P^{'}$ is the permutation matrix related to the n length cycle $[2, 3, 4, \ldots, n-1, n, 1]$[3] and $D_1^{'} = diag(a, 1, 1, \ldots, 1)$.

Thus, to find the k-NMDS (with $k = n-1$ and $k = n$) DLS matrices over $\mathbb{F}_{2^r}$, we need to check only for the DLS matrices in the set $\mathbb{D}^{'}(n, \mathbb{F}_{2^r})$.

From the discussion of Section 5.2.2, we know that if $\rho = [2, 3, 4, \ldots, n-1, n, 1]$ and D_2 has any two consecutive zero entries, then the DLS matrix of order n cannot be n-MDS. However, this result is not true for NMDS matrices. For example, consider the DLS matrix $B = DLS(\rho; D_1, D_2)$ of order 4 over $\mathbb{F}_{2^4}$/0x13, where $\rho = [2, 3, 4, 1]$, $D_1 = diag(1, 1, 1, 1)$, $D_2 = diag(1, \alpha, 0, 0)$ and $\alpha^4 + \alpha + 1 = 0$. Then it can be checked that B is 4-NMDS.

Thus, for finding k-NMDS DLS matrices with $k \in \{n-1, n\}$ and $\mathcal{K} = l$, we need to check for all the ${}^nC_l$ arrangements for the l nonzero elements in the diagonal of D_2. Thus, the search space for finding k-NMDS DLS matrices, with $k \in \{n-1, n\}$, over the field $\mathbb{F}_{2^r}$ has been reduced from $D(n) \cdot {}^nC_l \cdot (2^r)^{(n+l)}$ to ${}^nC_l \cdot (2^r)^{(1+l)}$. Then, by exhaustive search in the restricted domain, we have the results for the existence of k-NMDS DLS matrices over $\mathbb{F}_{2^4}$ and $\mathbb{F}_{2^8}$ for $n = 4, 5, 6, 7, 8$ listed in Table 6.1.

Table 6.1: k-NMDS DLS matrix of order n over the field $\mathbb{F}_{2^r}$ with $k = n-1$ and $k = n$ ("**DNE**" stands for does not exist).

Order n	k	$\mathcal{K} = 2$		$\mathcal{K} = 3$		$\mathcal{K} = 4$	
		over $\mathbb{F}_{2^4}$	**over** $\mathbb{F}_{2^8}$	**over** $\mathbb{F}_{2^4}$	**over** $\mathbb{F}_{2^8}$	**over** $\mathbb{F}_{2^4}$	**over** $\mathbb{F}_{2^8}$
4	3	Exists	Exists	–	–	–	–
	4	Exists	Exists	–	–	–	–
5	4	DNE	DNE	Exists	Exists	–	–
	5	DNE	DNE	Exists	Exists	–	–
6	5	DNE	DNE	Exists	Exists	–	–
	6	DNE	DNE	Exists	Exists	–	–
7	6	DNE	DNE	DNE	$\star$[a]	Exists	Exists
	7	DNE	DNE	DNE	$\star$	Exists	Exists
8	7	DNE	DNE	DNE	DNE	DNE	Exists
	8	DNE	DNE	DNE	DNE	DNE	Exists

[a]Over $\mathbb{F}_{2^8}$, we are unable to make a decision for $n = 7$ with $\mathcal{K} = 3$ since we were unable to perform an exhaustive search even in the restricted domain.

[3]By Remark 6.8, any n length cycle can be chosen for the set $\mathbb{D}^{'}(n, \mathbb{F}_{2^r})$.

6.3 Construction of Recursive NMDS Matrices from GDLS Matrices

In this section, we present some lightweight recursive NMDS matrices of orders $4, 5, 6, 7$, and 8 from the GDLS matrices, introduced in Section 5.3.

From Lemma 6.1, we know that if D_2 is singular, a DLS matrix $DLS(\rho; D_1, D_2)$ can never be k-NMDS for $k \leq n-2$. However, this result is not applicable to GDLS matrices. For instance, the GDLS matrix $M = GDLS(\rho_1, \rho_2; D_1, D_2)$ of order 7 with $\rho_1 = [6, 7, 4, 5, 2, 3, 1]$, $\rho_2 = [3, 2, 1, 4, 7, 6, 5]$, $D_1 = diag(1,\ 1,\ 1,\ 1,\ 1,\ 1,\ \alpha)$ and $D_2 = diag(1, 0, \alpha^2, 0, \alpha, 0, \alpha^2)$ is 5-NMDS, where α is a primitive element of the field $\mathbb{F}_{2^4}$ with $\alpha^4 + \alpha + 1 = 0$.

Since GDLS matrices have the potential to generate NMDS matrices with fewer iterations, we select them for constructing recursive NMDS matrices. To find recursive NMDS matrix, we begin with $k = n-2$, and if this does not yield a result, we increase the value of k.

From the definition of GDLS matrices, it can be observed that the size of the set of all GDLS matrices with $\mathcal{K} = l$ over the field $\mathbb{F}_{2^r}$ is $n! \cdot D(n) \cdot {}^nC_l \cdot (2^r)^{(n+l)}$, where $D(n)$ represents the number of derangements for n distinct objects. This size is extremely large, making an exhaustive search impractical for obtaining a k-NMDS matrix of order $n \geq 5$ from the GDLS matrices.

To minimize the search space, in most cases, we arbitrarily select ρ_1 as the n-cycle $[n, 1, 2, \ldots, n-1]$. However, it is important to note that there is no inherent advantage in choosing $\rho_1 = [n, 1, 2, \ldots, n-1]$ for obtaining a recursive NMDS matrix. If we change $\rho_1 = [n, 1, 2, \ldots, n-1]$ to any permutation from S_n, there is still a possibility of obtaining a recursive NMDS matrix.

Also, to find lightweight recursive NMDS matrices, we looked through the GDLS matrices of order n with $\mathcal{K} = \left\lceil \frac{n}{2} \right\rceil$ whose entries are from the set $\{1, \alpha, \alpha^{-1}, \alpha^2, \alpha^{-2}\}$, where α is a primitive element and a root of the constructing polynomial of the field $\mathbb{F}_{2^r}$. The search space for finding k-NMDS matrices of order $n \geq 5$ remains large, even when considering the set $\{1,\ \alpha,\ \alpha^{-1},\ \alpha^2,\ \alpha^{-2}\}$. Therefore, to obtain k-NMDS matrices of order $n = 5, 6, 7, 8$, we conduct a random search.

Also note that the implementation costs of the matrices presented in this section over a field are calculated by referring to the s-XOR count value of the corresponding field elements as provided in table of [TTKS18, App. B].

6.3.1 Construction of 4×4 Recursive NMDS matrices

In this section, we propose a GDLS matrix B of order 4 that yields a recursive NMDS matrix over the field $\mathbb{F}_{2^r}$ for $r \geq 1$. Based on Theorem 6.3, it is known that there are no k-NMDS matrices of order 4 with $\mathcal{K} = 1$ and $k \leq 4$ over a field of characteristic 2.

Therefore, to obtain recursive NMDS matrices of order 4, we must choose $\mathcal{K} \geq 2$. The proposed GDLS matrix is constructed by the permutations $\rho_1 = [2, 3, 4, 1]$, $\rho_2 = [1, 2, 3, 4]$ and diagonal matrices $D_1 = diag(1, 1, 1, 1)$, $D_2 = diag(0, 1, 0, 1)$.

$$B = GDLS(\rho_1, \rho_2; D_1, D_2) = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \tag{6.5}$$

The matrix B is a 3-NMDS matrix with a XOR count of $2 \cdot r = 2r$ over the field $\mathbb{F}_{2^r}$.

Lemma 6.3. *For k-NMDS matrix of orders 4 with $k \leq 4$, the lowest XOR count is $2r$ over the field $\mathbb{F}_{2^r}$.*

Proof. From Remark 6.5 and Theorem 6.2, we know that any matrix M of order 4 with $\mathcal{K} = 1$ cannot be k-NMDS for $k \leq 4$. Hence, we must have $\mathcal{K} \geq 2$. Therefore, we have $XOR(M) \geq 2 \cdot r$ over the field $\mathbb{F}_{2^r}$. □

Remark 6.9. *For $k \leq 4$, the proposed matrix B in (6.5) has the lowest XOR count among the k-NMDS matrices of order 4 over the field $\mathbb{F}_{2^r}$ for $r \geq 1$.*

6.3.2 Construction of 5×5 Recursive NMDS matrices

This section presents two GDLS matrices, A_1 and A_2, of order 5 that give NMDS matrices when raised to power 4 and 5, respectively, over the field $\mathbb{F}_{2^4}$. We also looked for GDLS matrices M of order 5 such that M^k is NMDS for $k \leq 3$ and $\mathcal{K} = 3$, but we were unable to find any over $\mathbb{F}_{2^4}$. Consider the GDLS matrices A_1 and A_2 of order 5 which are constructed as follows:

(i) A_1: $\rho_1 = [5, 1, 2, 3, 4], \rho_2 = [3, 2, 5, 4, 1]$, $D_1 = diag(1, 1, 1, 1, 1)$ and $D_2 = diag(0, \alpha, 0, 1, \alpha^{-1})$

(ii) A_2: $\rho_1 = [5, 1, 2, 3, 4], \rho_2 = [3, 4, 5, 1, 2]$, $D_1 = diag(1, 1, 1, 1, 1)$ and $D_2 = diag(0, 1, 0, 1, \alpha)$

$$A_1 = \begin{bmatrix} 0 & 1 & 0 & 0 & \alpha^{-1} \\ 0 & \alpha & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix} \quad A_2 = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & \alpha \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix}, \tag{6.6}$$

where α is a primitive element of $\mathbb{F}_{2^4}$ with $\alpha^4 + \alpha + 1 = 0$. It is easy to verify that the matrix A_1 is a 4-NMDS matrix with a XOR count of $(1+1) + 3 \cdot 4 = 14$ and A_2 is a 5-NMDS matrix with a XOR count of $1 + 3 \cdot 4 = 13$.
In Lemma 6.4, we discuss the lowest XOR count of recursive NMDS matrices of order $n \geq 5$. For this, we need the following result from [CK08].

Theorem 6.4. *[CK08] A matrix of order n, with 0 and 1 as entries, has a maximum branch number of* $\frac{2n+4}{3}$.

Lemma 6.4. *Given a recursive NMDS matrix B of orders $n \geq 5$, with $\mathcal{K} = l$, the lowest XOR count of B is $XOR(\beta) + l \cdot r$ over the field $\mathbb{F}_{2^r}$, where β ($\neq 1$) is a nonzero element in $\mathbb{F}_{2^r}$ with the lowest XOR count value in that field.*

Proof. An NMDS matrix of order n has branch number of n. Therefore, based on Theorem 6.4, it can be concluded that a matrix of order n, containing elements from the set $\{0, 1\} \subseteq \mathbb{F}_{2^r}$, cannot be NMDS when $n \geq 5$. If we take a matrix B with entries of 0 or 1, then the entries of B^k will remain in the set $\{0, 1\}$ for any power k. So B must have an element $\gamma \notin \{0, 1\}$. Therefore, $XOR(B) \geq XOR(\beta) + l \cdot r$, where β ($\neq 1$) is a nonzero element in $\mathbb{F}_{2^r}$ with the lowest XOR count value in that field. □

Remark 6.10. *Over the field $\mathbb{F}_{2^4}$, the matrix A_2 in (6.6) has the lowest XOR count among the 5-NMDS matrices of order 5 and $\mathcal{K} = 3$.*

6.3.3 Construction of 6×6 Recursive NMDS matrices

In this section, we introduce two lightweight GDLS matrices, B_1 and B_2, of order 6 with $\mathcal{K} = 3$. These matrices can be implemented with 14 and 13 XORs over the field $\mathbb{F}_{2^4}$, respectively, and yield NMDS matrices when raised to the power of 5 and 6, respectively. The matrices B_1 and B_2 of order 6 are constructed as follows:

(i) B_1: $\rho_1 = [6, 1, 2, 3, 4, 5]$, $\rho_2 = [1, 2, 3, 4, 5, 6]$, $D_1 = diag(1, 1, 1, 1, 1, 1)$ and $D_2 = diag(0, \alpha, 0, \alpha^{-1}, 0, 1)$

(ii) B_2: $\rho_1 = [6, 1, 2, 3, 4, 5], \rho_2 = [3, 4, 5, 2, 6, 1]$, $D_1 = diag(1, 1, 1, 1, 1, 1)$ and $D_2 = diag(0, \alpha, 0, 1, 0, 1)$

$$B_1 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & \alpha & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & \alpha^{-1} & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad B_2 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & \alpha & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \tag{6.7}$$

where α is a primitive element of $\mathbb{F}_{2^4}$ with $\alpha^4 + \alpha + 1 = 0$. It can be checked that the matrix B_1 is a 5-NMDS matrix with a XOR count of $(1 + 1) + 3 \cdot 4 = 14$ and B_2 is a 6-NMDS matrix with a XOR count of $1 + 3 \cdot 4 = 13$.

We also searched for GDLS matrices M of order 6 such that M^k is NMDS for $k \leq 4$ and $\mathcal{K} = 3$, but we could not find such matrices over $\mathbb{F}_{2^4}$.

Remark 6.11. *It is not possible to have elements with XOR count* 1 *in* $\mathbb{F}_{2^8}$ *due to the absence of trinomial irreducible polynomial of degree* 8 *over* $\mathbb{F}_2$ *[BKL16, Theorem 2]. However, it is possible to have elements with XOR count of* 1 *over rings.*

Consider the binary matrix $C = [[2], [3], [4], [5], [6], [7], [8], [1, 3]]$ *which is the companion matrix of* $x^8 + x^2 + 1$ *over* $\mathbb{F}_2$. *If we replace* α *by* C, *then the matrices* A_1 *and* A_2 *in (6.6) and* B_1 *and* B_2 *in (6.7) will be* 4*-NMDS,* 5*-NMDS,* 5*-NMDS, and* 6*-NMDS over* $GL(8, \mathbb{F}_2)$, *respectively. In addition, the implementation cost of* C *and* C^{-1} *is* 1 *XOR. Hence, the implementation cost of* A_1, A_2, B_1 *and* B_2 *over* $GL(8, \mathbb{F}_2)$ *are* $26, 25, 26$ *and* 25 *XORs, respectively.*

Remark 6.12. *Over the field* $\mathbb{F}_{2^4}$, *the matrix* B_2 *in (6.7) has the lowest XOR count among the* 6*-NMDS matrices of order* 6 *and* $\mathcal{K} = 3$.

6.3.4 Construction of 7×7 Recursive NMDS matrices

Now, we propose three GDLS matrices of order 7 that yield NMDS matrices over the field $\mathbb{F}_{2^4}$ for $\mathcal{K} = 4$. Consider the GDLS matrices B_1, B_2 and B_3 of order 7 which are constructed as follows:

(i) B_1: $\rho_1 = [6, 7, 4, 5, 2, 3, 1]$, $\rho_2 = [3, 2, 1, 4, 7, 6, 5]$, $D_1 = diag(1,\ 1,\ 1,\ 1,\ 1,\ 1,\ \alpha)$ and $D_2 = diag(1, 0, \alpha^2, 0, \alpha, 0, \alpha^2)$

(ii) B_2: $\rho_1 = [7, 1, 2, 3, 4, 5, 6]$, $\rho_2 = [6, 7, 5, 4, 1, 3, 2]$, $D_1 = diag(1, \alpha^{-1}, \alpha, 1, \alpha, \alpha, 1)$ and $D_2 = diag(0, 1, 0, 1, 0, 1, 1)$

(iii) B_3: $\rho_1 = [7, 1, 2, 3, 4, 5, 6]$, $\rho_2 = [5, 2, 6, 7, 3, 1, 4]$, $D_1 = diag(1,\ 1,\ 1,\ 1,\ 1,\ 1,\ 1)$ and $D_2 = diag(0, \alpha^{-1}, 0, 1, 0, \alpha^{-1}, 1)$

$$B_1 = \begin{bmatrix} 0 & 0 & \alpha^2 & 0 & 0 & 0 & \alpha \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & \alpha^2 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & \alpha & 0 & 0 \end{bmatrix} B_2 = \begin{bmatrix} 0 & \alpha^{-1} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \alpha & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & \alpha & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \alpha & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} B_3 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & \alpha^{-1} & 0 \\ 0 & \alpha^{-1} & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}, \quad (6.8)$$

where α is a primitive element of $\mathbb{F}_{2^4}$ with $\alpha^4 + \alpha + 1 = 0$. It can be verified that matrix B_1 is a 5-NMDS matrix with an XOR count of $(1 + 2 + 1 + 2) + 4 \cdot 4 = 22$, B_2 is a 6-NMDS matrix with an XOR count of $(1 + 1 + 1 + 1) + 4 \cdot 4 = 20$, and B_3 is a 7-NMDS matrix with an XOR count of $(1 + 1) + 4 \cdot 4 = 18$.

Remark 6.13. *If we replace α by C (the binary matrix in Remark 6.11), then the matrices B_1, B_2 and B_3 in (6.8) will be 5-NMDS, 6-NMDS and 7-NMDS over $GL(8, \mathbb{F}_2)$, respectively. The binary matrix C^2 can be implemented with 2 XORs. Hence, B_1, B_2 and B_3 can be implemented with $38, 36$ and 34 XORs, respectively, over $GL(8, \mathbb{F}_2)$.*

Remark 6.14. *We know that in a DLS matrix $M = DLS(\rho_1; D_1, D_2)$ of order $n \geq 2$, if D_2 is singular, then M^k cannot be NMDS for $k \leq n - 2$. However, the result is not true for GDLS matrices. For example the matrix B_1 of order 7 in (6.8) is 5-NMDS.*

6.3.5 Construction of 8×8 Recursive NMDS matrices

As 4 and 8 are the most commonly used diffusion layer matrix sizes, we look for a k-NMDS GDLS matrix of order 8 over $\mathbb{F}_{2^4}$. However, we were unable to find a GDLS matrix of order 8, which corresponds to 7-NMDS or 8-NMDS over $\mathbb{F}_{2^4}$. Nonetheless, we have proposed two GDLS matrices of order 8 that yield NMDS matrices over the field $\mathbb{F}_{2^8}$ with $\mathcal{K} = 4$. Consider the GDLS matrices B_1 and B_2 of order 8 which are constructed as follows:

(i) B_1 : $\rho_1 = [2, 3, 4, 5, 6, 7, 8, 1]$, $\rho_2 = [3, 8, 5, 2, 1, 4, 6, 7]$, $D_1 = diag(1,\ 1,\ 1,\ 1,\ 1,\ \alpha^{-2}, 1, 1)$ and $D_2 = diag(1, 0, \alpha, 0, 1, 0, \alpha^{-1}, 0)$

(ii) B_2 : $\rho_1 = [2, 3, 4, 5, 6, 7, 8, 1]$, $\rho_2 = [3, 8, 5, 2, 1, 4, 6, 7]$, $D_1 = diag(1, 1, 1, \alpha^2, 1, 1, 1, 1)$ and $D_2 = diag(\alpha, 0, 1, 0, 1, 0, 1, 0)$

$$B_1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \alpha & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & \alpha^{-1} & 0 \\ 0 & 0 & 0 & 0 & 0 & \alpha^{-2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \quad B_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \alpha & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & \alpha^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}, \tag{6.9}$$

where α is a primitive element of $\mathbb{F}_{2^8}$ with $\alpha^8+\alpha^7+\alpha^6+\alpha+1=0$. The matrix B_1 is a 7-NMDS matrix with a XOR count of $(4+3+3)+4\cdot 8 = 42$ and B_2 is a 8-NMDS matrix with a XOR count of $(3+4)+4\cdot 8 = 39$.

Remark 6.15. *Consider the binary matrix* $C_8 = [[8],[1,2],[2,8],[3],[4],[5],[6],[7]]$ *whose minimal polynomial is* $x^8+x^7+x^2+x+1$. *Then by replacing* α *by* C_8, *the matrices* B_1, *and* B_2 *in (6.9) will be* 7*-NMDS and* 8*-NMDS over* $GL(8,\mathbb{F}_2)$, *respectively. In addition, the implementation cost of* C_8 *is* 2 *XORs. Also,* C_8^{-1}, C_8^2 *and* C_8^{-2} *can be implemented with* 2, 4 *and* 4 *XORs respectively. Hence,* B_1 *and* B_2 *can be implemented with* 40 *and* 38 *XORs, respectively, over* $GL(8,\mathbb{F}_2)$.

Until now, we have discussed NMDS matrices in a recursive setup. While these matrices have a low hardware cost, they do require some clock cycles. To use recursive NMDS (say, k-NMDS) matrices in an unrolled implementation, we have to add k copies of the matrix to the circuit in sequence, which may increase the cost of the diffusion layer. This makes recursive NMDS matrices less suitable for block ciphers that operate within a single clock cycle, such as PRINCE [BCG+12] and MANTIS [BJK+16]. From the next section on, we will discuss nonrecursive constructions of NMDS matrices.

6.4 Construction of Nonrecursive NMDS Matrices

The construction of nonrecursive MDS matrices is typically based on specific matrix types such as circulant matrices, Hadamard matrices, Cauchy matrices, Vandermonde matrices, and Toeplitz matrices. A brief summary of such constructions is presented in Chapter 3. Circulant and Hadamard matrices of order n can have at most n distinct elements; thus, these matrices are used to reduce the search space. Furthermore, it is

Table 6.2: Comparison of recursive NMDS matrices of order n.

Order n	Input	Iterations	Field/Ring	XOR count	References
4	4-bit	34	$M_4(\mathbb{F}_2)$	**1**	[LW21]
4	4-bit	16	$M_4(\mathbb{F}_2)$	2	[LW21]
4	4-bit	10	$M_4(\mathbb{F}_2)$	3	[LW21]
4	4-bit	7	$M_4(\mathbb{F}_2)$	4	[LW21]
4	4-bit	5	$M_4(\mathbb{F}_2)$	7	[LW21]
4	4-bit	3	$M_4(\mathbb{F}_2)$	8	[LW21]
4	4-bit	3	$\mathbb{F}_{2^4}$	8	Section 6.3.1
4	4-bit	**2**	$M_4(\mathbb{F}_2)$	12	[LW21]
4	8-bit	66	$M_8(\mathbb{F}_2)$	**1**	[LW21]
4	8-bit	34	$M_8(\mathbb{F}_2)$	2	[LW21]
4	8-bit	16	$M_8(\mathbb{F}_2)$	4	[LW21]
4	8-bit	10	$M_8(\mathbb{F}_2)$	6	[LW21]
4	8-bit	7	$M_8(\mathbb{F}_2)$	8	[LW21]
4	8-bit	3	$M_8(\mathbb{F}_2)$	16	[LW21]
4	8-bit	3	$\mathbb{F}_{2^8}$	16	Section 6.3.1
4	8-bit	**2**	$M_4(\mathbb{F}_2)$	24	[LW21]
5	4-bit	86	$M_8(\mathbb{F}_2)$	**1**	[LW21]
5	4-bit	46	$M_8(\mathbb{F}_2)$	2	[LW21]
5	4-bit	20	$M_8(\mathbb{F}_2)$	3	[LW21]
5	4-bit	15	$M_8(\mathbb{F}_2)$	4	[LW21]
5	4-bit	8	$M_8(\mathbb{F}_2)$	8	[LW21]
5	4-bit	5	$\mathbb{F}_{2^4}$/0x13	13	Section 6.3.2
5	4-bit	**4**	$\mathbb{F}_{2^4}$/0x13	14	Section 6.3.2
5	8-bit	120	$M_8(\mathbb{F}_2)$	**1**	[LW21]
5	8-bit	86	$M_8(\mathbb{F}_2)$	2	[LW21]
5	8-bit	46	$M_8(\mathbb{F}_2)$	4	[LW21]
5	8-bit	20	$M_8(\mathbb{F}_2)$	6	[LW21]
5	8-bit	15	$M_8(\mathbb{F}_2)$	8	[LW21]
5	8-bit	8	$M_8(\mathbb{F}_2)$	16	[LW21]
5	8-bit	5	$GL(8,\mathbb{F}_2)$	25	Remark 6.11
5	8-bit	**4**	$GL(8,\mathbb{F}_2)$	26	Remark 6.11
6	4-bit	6	$\mathbb{F}_{2^4}$/0x13	**13**	Section 6.3.3
6	4-bit	**5**	$\mathbb{F}_{2^4}$/0x13	14	Section 6.3.3
6	8-bit	6	$GL(8,\mathbb{F}_2)$	**25**	Remark 6.11
6	8-bit	**5**	$GL(8,\mathbb{F}_2)$	26	Remark 6.11
7	4-bit	7	$\mathbb{F}_{2^4}$/0x13	**18**	Section 6.3.4
7	4-bit	6	$\mathbb{F}_{2^4}$/0x13	20	Section 6.3.4
7	4-bit	**5**	$\mathbb{F}_{2^4}$/0x13	22	Section 6.3.4
7	8-bit	7	$GL(8,\mathbb{F}_2)$	**34**	Remark 6.13
7	8-bit	6	$GL(8,\mathbb{F}_2)$	36	Remark 6.13
7	8-bit	**5**	$GL(8,\mathbb{F}_2)$	38	Remark 6.13
8	8-bit	8	$\mathbb{F}_{2^8}$/0x1c3	39	Section 6.3.5
8	8-bit	8	$GL(8,\mathbb{F}_2)$	**38**	Section 6.3.5
8	8-bit	**7**	$\mathbb{F}_{2^8}$/0x1c3	42	Section 6.3.5
8	8-bit	**7**	$GL(8,\mathbb{F}_2)$	40	Section 6.3.5

worth noting that circulant matrices offer the advantage of being adaptable for implementation in both round-based and serialized implementations [LS16]. In [LW17], the authors have studied the construction of NMDS matrices using circulant and Hadamard matrices and present some generic NMDS matrices of order n for the range of $5 \leq n \leq 9$.
In the context of implementing block ciphers, we know that if an efficient matrix M used in encryption is involutory, then its inverse $M^{-1} = M$ applied for decryption will also be efficient. Therefore, it is particularly important to locate NMDS matrices that are also involutory. In this regard, Li et al. [LW17] show that for $n > 4$, no circulant matrices of order n over $\mathbb{F}_{2^r}$ can simultaneously be involutory and NMDS. We recall it in the following theorem.

Theorem 6.5. *[LW17] Over the field $\mathbb{F}_{2^r}$, circulant involutory matrices of order $n > 4$ are not NMDS.*

Remark 6.16. *For $n < 4$, there may exist circulant involutory NMDS matrices over $\mathbb{F}_{2^r}$. For example, the circulant matrix $Circ(0, 1, 1, 1)$ of order 4 is both involutory and NMDS over the field $\mathbb{F}_{2^r}$.*

Remark 6.17. *According to Lemma 3.17, the above result is also true for circulant MDS matrices of order n with a modified lower bound of $n \geq 3$.*

For symmetric cryptography, having an orthogonal matrix as the linear diffusion layer simplifies decryption because the transpose of an orthogonal matrix is its inverse. This makes orthogonal matrices ideal for constructing the linear diffusion layer. Matrices of order 2^n are particularly important in cryptography. However, as stated in Lemma 3.14, for $n \geq 2$, we know that any orthogonal circulant matrix of order 2^n over the field $\mathbb{F}_{2^r}$ is not MDS. But circulant MDS matrices of different orders may be orthogonal over $\mathbb{F}_{2^r}$ (see Remark 3.24).

Remark 6.18. *NMDS circulant orthogonal matrices of any order may exist over the field $\mathbb{F}_{2^r}$. For example, consider the circulant matrices $Circ(0, \alpha^3 + \alpha + 1, \alpha^3 + \alpha^2 + \alpha, \alpha^3 + 1, \alpha^3 + \alpha^2 + 1)$, $Circ(0, 1, \alpha, \alpha^2 + \alpha + 1, \alpha^3 + \alpha + 1, \alpha^3 + \alpha^2 + \alpha)$, and $Circ(0, 1, \alpha, \alpha + 1, \alpha + 1, \alpha^3 + \alpha^2 + \alpha + 1, 1, \alpha^3 + \alpha^2)$ of order $5, 6$, and 8, respectively, where α is a primitive element of $\mathbb{F}_{2^4}$ with $\alpha^4 + \alpha + 1 = 0$. It can be checked that these matrices are both orthogonal and NMDS.*

Note that a left-circulant matrix is symmetric; consequently, if the matrix is orthogonal, then it is involutory, and vice versa.

Remark 6.19. *From Lemma 2.11, we know that if M is an NMDS matrix, then for any permutation matrix P, PM is also an NMDS matrix. Additionally, as per Remark 6.18, it is possible to obtain a circulant NMDS matrix $M = Circ(x_1, x_2, \ldots, x_n)$ of any order n over $\mathbb{F}_{2^r}$ that is orthogonal. Now, we know that $PM = l\text{-}Circ(x_1, x_2, \ldots, x_n)$, where P is the permutation matrix given in (2.1). Also, since M is orthogonal and P is a permutation matrix, it follows that*

$$(PM)^T = M^T P^T = M^{-1} P^{-1} = (PM)^{-1}.$$

Thus, by the multiplication PM will not alter NMDS and orthogonality property for NMDS matrices. Consequently, the resulting matrix $PM = l\text{-}Circ(x_1, x_2, \ldots, x_n)$ will be both orthogonal and NMDS, making it an involutory NMDS matrix. Therefore, NMDS left-circulant involutory (orthogonal) matrices of any order may exist over the field $\mathbb{F}_{2^r}$.

The absence of any zero entries is a necessary condition for matrices such as Hadamard, circulant and left-circulant matrices to be MDS. Therefore, these matrices result in a high implementation cost due to $\mathcal{K} = n(n-1)$. Having zero entries (with a maximum of one zero per row or column) does not affect the NMDS property of these matrices, leading to a low implementation cost with $\mathcal{K} = n(n-2)$. Taking advantage of this, the authors in [LW17] provided some generic lightweight involutory NMDS matrices of order 8 from Hadamard matrices.

Theorem 6.6. *For a Hadamard, circulant, or left-circulant NMDS matrix of order n over $\mathbb{F}_{2^r}$ with $n \geq 5$, the XOR count is at least $XOR(\beta) \cdot n + n(n-2) \cdot r$ over the field $\mathbb{F}_{2^r}$, where β ($\neq 1$) is a nonzero element in $\mathbb{F}_{2^r}$ with the lowest XOR count value in that field.*

Proof. An NMDS matrix B of order n has branch number of n. Therefore, based on Theorem 6.4, it can be concluded that a matrix of order n, containing elements from the set $\{0, 1\} \subseteq \mathbb{F}_{2^r}$, cannot be NMDS when $n \geq 5$. This means that B must contain an element $\gamma \notin \{0, 1\}$. Additionally, for an NMDS matrix, we must have $\mathcal{K} \geq n(n-2)$. Also, each row in a Hadamard, circulant, or left-circulant matrix is a rearrangement of the first row. Hence, for these matrices to be NMDS over the field $\mathbb{F}_{2^r}$, the minimum XOR count must be $XOR(\beta) \cdot n + n(n-2) \cdot r$. □

The lowest XOR count value (of an element) in the field $\mathbb{F}_{2^4}$ is one, which allows us to obtain the lowest possible XOR count of Hadamard, circulant, or left-circulant NMDS matrices of various orders over $\mathbb{F}_{2^4}$ as shown in Table 6.3.

Table 6.3: Lowest possible XOR count of Hadamard, circulant, or left-circulant NMDS matrices of order n over $\mathbb{F}_{2^4}$.

order n	**5**	**6**	**7**	**8**
Lowest XOR count	65	102	147	200

The use of Toeplitz matrices for the construction of MDS matrices has been explored in the literature [SS16, SS17], and we will discuss them for the construction of NMDS matrices.

Theorem 6.7. *Over the field $\mathbb{F}_{2^r}$, Toeplitz involutory matrices of order $n > 4$ are not NMDS.*

Proof. Let $M = Toep(x_1, x_2, \ldots, x_n;\ y_1, y_2, \ldots, y_{n-1})$ be a Toeplitz matrix (as in Definition 2.32) of order n which is both involutory and NMDS over the field $\mathbb{F}_{2^r}$, where $n > 4$. We will examine two scenarios: when n is even and when n is odd.

Case 1: n is even.
In an NMDS matrix, there may be a zero entry. So this case splits into two subcases: $x_n \neq 0$ and $x_n = 0$.
Case 1.1: When $x_n \neq 0$.
The $(n-1)$-th element in the 1st row of M^2 is

$$\begin{aligned}(M^2)_{1,n-1} &= M_{row(1)} \cdot M_{column(n-1)} \\ &= x_1x_{n-1} + x_2x_{n-2} + \cdots + x_{\frac{n}{2}}x_{\frac{n}{2}} + \cdots + x_{n-1}x_1 + x_ny_1 \\ &= x_{\frac{n}{2}}^2 + x_ny_1.\end{aligned}$$

Since M is involutory, we have $(M^2)_{1,n-1} = 0$. Therefore, from above we have

$$x_{\frac{n}{2}}^2 + x_ny_1 = 0 \tag{6.10}$$

$$\implies y_1 = x_{\frac{n}{2}}^2 x_n^{-1}. \tag{6.11}$$

We have

$$\begin{aligned}(M^2)_{1,n-2} &= M_{row(1)} \cdot M_{column(n-2)} \\ &= x_1x_{n-2} + x_2x_{n-3} + \cdots + x_{\frac{n-2}{2}}x_{\frac{n}{2}} + x_{\frac{n}{2}}x_{\frac{n-2}{2}} + \cdots \\ &\quad + x_{n-3}x_2 + x_{n-2}x_1 + x_{n-1}y_1 + x_ny_2 \\ &= x_{n-1}y_1 + x_ny_2.\end{aligned}$$

Also, $(M^2)_{1,n-2} = 0$, which results in

$$x_{n-1}y_1 + x_ny_2 = 0 \tag{6.12}$$

Now, from Equation 6.11 and Equation 6.12, we have

$$y_2 = x_{\frac{n}{2}}^2 x_{n-1} x_n^{-2}. \tag{6.13}$$

Also, from $(M^2)_{3,n-1} = 0$, we have

$$\begin{aligned}
& x_{\frac{n-2}{2}}^2 + x_{n-1} y_2 = 0 \\
\implies & x_{\frac{n-2}{2}}^2 + x_{n-1} \cdot x_{\frac{n}{2}}^2 x_{n-1} x_n^{-2} = 0 \qquad \text{[From Equation 6.13]} \\
\implies & x_{\frac{n-2}{2}}^2 x_n^2 = x_{\frac{n}{2}}^2 x_{n-1}^2 \\
\implies & x_{\frac{n-2}{2}} x_n = x_{\frac{n}{2}} x_{n-1} \qquad \text{[Since characteristic of } \mathbb{F}_{2^r} \text{ is 2]} \qquad (6.14)
\end{aligned}$$

Now consider the input vector $v = [0, 0, \ldots, \underbrace{x_n}_{\frac{n}{2}\text{-th}}, 0, \ldots, x_{\frac{n}{2}}]^T$ of M. Therefore, we have

$$\begin{aligned}
M \cdot v &= [x_{\frac{n}{2}} x_n + x_n x_{\frac{n}{2}},\ x_{\frac{n-2}{2}} x_n + x_{n-1} x_{\frac{n}{2}},\ *, \ldots, *, \underbrace{y_1 x_n + x_{\frac{n}{2}}^2}_{(\frac{n}{2}+1)\text{-th}}, *, \ldots, *]^T \\
&= [0, 0, *, \ldots, \underbrace{0}_{(\frac{n}{2}+1)\text{-th}}, *, \ldots, *]^T,
\end{aligned}$$

where $*$ denotes some entry may or may not be zero. Here the second and $(\frac{n}{2}+1)$-th coordinates of $M \cdot v$ are zero by Equation 6.14 and Equation 6.10, respectively. Thus, the sum of nonzero elements of input vector (v) and output vector $(M \cdot v)$ is $\leq 2 + (n-3) < n$ i.e. branch number of $M < n$. This contradicts that M is NMDS.

Case 1.2: When $x_n = 0$.

If $x_n = 0$, then from Equation 6.10, we conclude that $x_{\frac{n}{2}}^2 = 0$ which implies $x_{\frac{n}{2}} = 0$. Therefore, the Toeplitz matrix M has two zero entries in its first row, which contradicts the fact that M is NMDS.

Case 2: n is odd.

The n-th element in the 1st row of M^2 is

$$\begin{aligned}
(M^2)_{1,n} &= M_{row(1)} \cdot M_{column(n)} \\
&= x_1 x_n + x_2 x_{n-1} + \cdots + x_{\frac{n+1}{2}} x_{\frac{n+1}{2}} + \cdots + x_{n-1} x_2 + x_n x_1 \\
&= x_{\frac{n+1}{2}}^2.
\end{aligned}$$

Also, we have $(M^2)_{2,n-1} = x_{\frac{n-1}{2}}^2$. Therefore, since M is involutory, it follows that $(M^2)_{1,n} = (M^2)_{2,n-1} = 0$, implying that $x_{\frac{n-1}{2}} = x_{\frac{n+1}{2}} = 0$. This means that M has two zero entries in its first row, which contradicts that M is an NMDS matrix. Hence, the proof. □

Remark 6.20. *Circulant matrices are a particular type of Toeplitz matrices, and*

thus, from Remark 6.16, we can say that for $n < 4$, there may exist Toeplitz involutory NMDS matrices over $\mathbb{F}_{2^r}$.

Remark 6.21. *From Theorem 3.11, we know that for $n \geq 2$, any orthogonal Toeplitz matrix of order 2^n over the field $\mathbb{F}_{2^r}$ is not MDS. However, this result does not hold for NMDS matrices. Circulant matrices are a particular type of Toeplitz matrices, and thus, from Remark 6.18, we can say that Toeplitz orthogonal NMDS matrices of any order may exist over the field $\mathbb{F}_{2^r}$.*

Hankel matrices are symmetric and may be described by their first row and last column. Thus, an involutory (orthogonal) Hankel matrix is orthogonal (involutory).

Remark 6.22. *From Theorem 3.12, we know that for $n \geq 2$, any involutory (orthogonal) Hankel matrix of order 2^n over the field $\mathbb{F}_{2^r}$ is not MDS. However, this result does not hold for NMDS matrices. Left-circulant matrices are a particular type of Hankel matrices, and thus, from Remark 6.19, we can say that Hankel involutory (orthogonal) NMDS matrices of any order may exist over the field $\mathbb{F}_{2^r}$.*

We close this section by presenting Table 6.4, which compares the involutory and orthogonal properties of MDS and NMDS matrices constructed from the circulant, left-circulant, Toeplitz and Hankel families.

Table 6.4: Comparison of involutory and orthogonal properties of MDS and NMDS matrices over a finite field $\mathbb{F}_{2^r}$ ("**DNE**" stands for does not exist).

Type	**Property**	**Dimension**	**MDS**	**NMDS**
Circulant	Involutory	$n \times n$	DNE	DNE
	Orthogonal	$2^n \times 2^n$	DNE	may exist
		$2n \times 2n$	may exist	may exist
		$(2n+1)\times(2n+1)$	may exist	may exist
left-Circulant	Involutory	$2^n \times 2^n$	DNE	may exist
		$2n \times 2n$	may exist	may exist
		$(2n+1)\times(2n+1)$	may exist	may exist
Toeplitz	Involutory	$n \times n$	DNE	DNE
	Orthogonal	$2^n \times 2^n$	DNE	may exist
		$2n \times 2n$	may exist	may exist
		$(2n+1)\times(2n+1)$	may exist	may exist
Hankel	Involutory	$2^n \times 2^n$	DNE	may exist
		$2n \times 2n$	may exist	may exist
		$(2n+1)\times(2n+1)$	may exist	may exist

6.5 Construction of Nonrecursive NMDS Matrices from GDLS Matrices

Constructing NMDS matrices from circulant, left-circulant, Hadamard, Toeplitz or Hankel matrices of order n may result in a high implementation cost due to the requirement of having $\mathcal{K} \geq n(n-2)$. To address this issue, in this section, we present some lightweight nonrecursive NMDS matrices through the composition of various GDLS matrices, similar to the method used by the authors in [SM21] for constructing MDS matrices.

To minimize the search space, in most cases, we arbitrarily select ρ_1 as the n-cycle $[n,\ 1,\ 2,\ \ldots,\ n-1]$. However, it is important to note that there is no inherent advantage in choosing $\rho_1 = [n, 1, 2, \ldots, n-1]$ for obtaining an NMDS matrix. If we change $\rho_1 = [n, 1, 2, \ldots, n-1]$ to any permutation from S_n, there is still a possibility of obtaining an NMDS matrix.

To search for lightweight nonrecursive NMDS matrices, we examine GDLS matrices of order n with $\mathcal{K} = \left\lceil \frac{n}{2} \right\rceil$ and entries from the set $\{1, \alpha, \alpha^{-1}, \alpha^2, \alpha^{-2}\}$, where α is a primitive element and a root of the constructing polynomial of the field $\mathbb{F}_{2^r}$. The search space for finding nonrecursive NMDS matrices of order $n \geq 5$ remains large, even when considering the set $\{1, \alpha, \alpha^{-1}, \alpha^2, \alpha^{-2}\}$. Therefore, to obtain nonrecursive NMDS matrices of order $n = 5, 6, 7, 8$, we conduct a random search. In addition, to construct nonrecursive NMDS matrices of order n, we typically chose $n-2$ GDLS matrices of the same structure. If this does not yield result, we use $n-1$ matrices instead.

Also note that the implementation costs of the matrices presented in this section over a field are calculated by referring to the s-XOR count value of the corresponding field elements as provided in table of [TTKS18, App. B]. In Table 6.5, we compare our results for nonrecursive NMDS matrices with the existing results.

6.5.1 Construction of 4×4 nonrecursive NMDS matrices

From Remark 6.9, we know that the matrix B given in 6.5 has the lowest XOR count among all k-NMDS matrices with $k \leq 4$ over $\mathbb{F}_{2^r}$. The proposed GDLS matrix is 3-NMDS over a field $\mathbb{F}_{2^r}$. Therefore, we can obtain a nonrecursive NMDS matrix of order 4 by composing the matrix B with itself three times. This results in an implementation cost of $3 \cdot (2 \cdot r) = 6r$ over a field $\mathbb{F}_{2^r}$.

Table 6.5: Comparison of nonrecursive NMDS matrices of order n.

Order n	Input	**Field/Ring**	**XOR count**	**References**
4	4-bit	$\mathbb{F}_{2^4}$	24	[SM21]
4	4-bit	$\mathbb{F}_{2^4}$	24	Section 6.5.1
4	8-bit	$\mathbb{F}_{2^8}$	48	[SM21]
4	8-bit	$\mathbb{F}_{2^8}$	48	Section 6.5.1
5	4-bit	$\mathbb{F}_{2^4}$/0x13	65	[LW17]
5	4-bit	$\mathbb{F}_{2^4}$/0x13	**50**	Section 6.5.2
5	8-bit	$\mathbb{F}_{2^8}$/0x11b	130	[LW17]
5	8-bit	$GL(8, \mathbb{F}_2)$	**98**	Remark 6.23
6	4-bit	$\mathbb{F}_{2^4}$/0x13	108	[LW17]
6	4-bit	$\mathbb{F}_{2^4}$/0x13	**65**	Section 6.5.3
6	8-bit	$\mathbb{F}_{2^8}$/0x11b	216	[LW17]
6	8-bit	$GL(8, \mathbb{F}_2)$	**125**	Remark 6.24
7	4-bit	$\mathbb{F}_{2^4}$/0x13	154	[LW17]
7	4-bit	$\mathbb{F}_{2^4}$/0x13	**96**	Section 6.5.4
7	8-bit	$\mathbb{F}_{2^8}$/0x11b	308	[LW17]
7	8-bit	$GL(8, \mathbb{F}_2)$	**176**	Remark 6.25
8	4-bit	$\mathbb{F}_{2^4}$/0x13	216	[LW17]
8	4-bit	$\mathbb{F}_{2^4}$/0x13	108	[SM21]
8	4-bit	$\mathbb{F}_{2^4}$/0x13	108	Section 6.5.5
8	8-bit	$\mathbb{F}_{2^8}$/0x11b	432	[LW17]
8	8-bit	$GL(8, \mathbb{F}_2)$	204	[SM21]
8	8-bit	$GL(8, \mathbb{F}_2)$	204	Remark 6.26

6.5.2 Construction of 5×5 nonrecursive NMDS matrices

In this section, we propose three GDLS matrices, B_1, B_2 and B_3, which are constructed by the permutations $\rho_1 = [5, 1, 2, 3, 4]$, $\rho_2 = [3, 2, 5, 4, 1]$ and the following diagonal matrices.

(i) B_1: ρ_1, ρ_2, $D_1 = diag(1, 1, 1, 1, 1)$ and $D_2 = diag(0, \alpha, 0, 1, 1)$

(ii) B_2: ρ_1, ρ_2, $D_1 = diag(1, 1, 1, 1, 1)$ and $D_2 = diag(0, 1, 0, 1, 1)$

(iii) B_3: ρ_1, ρ_2, $D_1 = diag(1, 1, 1, 1, 1)$ and $D_2 = diag(0, 1, 0, \alpha, 1)$,

where α is a primitive element of $\mathbb{F}_{2^4}$ with $\alpha^4 + \alpha + 1 = 0$. Using these three GDLS matrices, we propose a 5×5 NMDS matrix as follows:

$$M = B_2B_3B_1B_2 = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \alpha & 1 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 & 0 & 1 \\ 0 & \alpha & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$
$$= \begin{bmatrix} 1 & \alpha+1 & \alpha+1 & 0 & 1 \\ 1 & \alpha & \alpha & 1 & 0 \\ \alpha & 1 & 0 & \alpha & \alpha+1 \\ \alpha+1 & 0 & 1 & \alpha & \alpha+1 \\ 0 & \alpha+1 & \alpha & 1 & 1 \end{bmatrix}. \tag{6.15}$$

Now, $XOR(M) = XOR(B_1) + 2 \cdot XOR(B_2) + XOR(B_3)$. Therefore, M can be implemented with $(1 + 3 \cdot 4) + 2 \cdot (0 + 3 \cdot 4) + (1 + 3 \cdot 4) = 50$ XORs over the field $\mathbb{F}_{2^4}$/0x13.

Remark 6.23. *As discussed in Remark 6.11, if α is replaced with C, the matrix M from (6.15) will be NMDS over $GL(8, \mathbb{F}_2)$, with an implementation cost of $(1 + 3 \cdot 8) + 2 \cdot (0 + 3 \cdot 8) + (1 + 3 \cdot 8) = 98$ XORs.*

6.5.3 Construction of 6×6 nonrecursive NMDS matrices

In this section, we propose a lightweight 6×6 NMDS matrix M that can be implemented with 65 XORs over the field $\mathbb{F}_{2^4}$. The matrix M is constructed from three GDLS matrices, B_1, B_2, and B_3, of order 6, as $M = B_2^2B_1B_3B_2$. These GDLS matrices are constructed using the permutations $\rho_1 = [6, 1, 2, 3, 4, 5]$ and $\rho_2 = [5, 6, 1, 2, 3, 4]$ and by the following diagonal matrices as follows:

(i) B_1: ρ_1, ρ_2, $D_1 = diag(1, 1, 1, \alpha, 1, \alpha)$ and $D_2 = diag(0, \alpha, 0, 1, 0, 1)$

(ii) B_2: ρ_1, ρ_2, $D_1 = diag(1, 1, 1, 1, 1, 1)$ and $D_2 = diag(0, 1, 0, 1, 0, 1)$

(iii) B_3: ρ_1, ρ_2, $D_1 = diag(\alpha, 1, 1, \alpha^{-1}, 1, 1)$ and $D_2 = diag(0, 1, 0, 1, 0, 1)$

$$B_1 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & \alpha & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & \alpha \\ 1 & \alpha & 0 & 0 & 0 & 0 \end{bmatrix} \quad B_2 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \quad B_3 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & \alpha^{-1} & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ \alpha & 1 & 0 & 0 & 0 & 0 \end{bmatrix}, \tag{6.16}$$

where α is a primitive element of $\mathbb{F}_{2^4}$ with $\alpha^4 + \alpha + 1 = 0$. Now it can be checked that M is NMDS over $\mathbb{F}_{2^4}$/0x13 with an implementation cost of 65 XORs, calculated as $XOR(M) = XOR(B_1) + 3 \cdot XOR(B_2) + XOR(B_3) = (1 + 1 + 1 + 3 \cdot 4) + 3 \cdot (0 + 3 \cdot 4) + (1 + 1 + 3 \cdot 4) = 65$.

Remark 6.24. *As discussed in Remark 6.11, if α is replaced with C, the matrix M constructed from B_1, B_2 and B_3 in (6.16) will be NMDS over $GL(8, \mathbb{F}_2)$, with an implementation cost of* 125 *XORs.*

6.5.4 Construction of 7×7 nonrecursive NMDS matrices

This section presents three GDLS matrices, B_1, B_2, and B_3, of order 7. These matrices are constructed using the permutations $\rho_1 = [6, 7, 4, 5, 2, 3, 1]$ and $\rho_2 = [3, 2, 1, 4, 7, 6, 5]$, along with specific diagonal matrices as follows:

(i) B_1: ρ_1, ρ_2, $D_1 = diag(1, \alpha^{-1}, 1, \alpha^{-2}, 1, \alpha^2, 1)$ and $D_2 = diag(1, 0, 1, 0, 1, 0, 1)$

(ii) B_2: ρ_1, ρ_2, $D_1 = diag(1, 1, 1, 1, 1, 1, 1)$ and $D_2 = diag(1, 0, 1, 0, 1, 0, 1)$

(iii) B_3: ρ_1, ρ_2, $D_1 = diag(1, 1, 1, 1, 1, 1, \alpha^{-1})$ and $D_2 = diag(1, 0, 1, 0, \alpha^{-2}, 0, 1)$

$$B_1 = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & \alpha^2 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \alpha^{-2} & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \alpha^{-1} & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \quad B_2 = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \quad B_3 = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & \alpha^{-1} \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & \alpha^{-2} & 0 & 0 \end{bmatrix}, \tag{6.17}$$

where α is a primitive element in $\mathbb{F}_{2^4}$ with $\alpha^4+\alpha+1 = 0$. Using these three GDLS matrices, we propose a 7×7 matrix M given by $M = B_3B_1^2B_3B_2$. It can be verified that M is an NMDS matrix over $\mathbb{F}_{2^4}$/0x13 with an implementation cost of 96 XORs, which is calculated as $XOR(M) = 2\cdot(1+2+2+4\cdot4)+(0+4\cdot4)+2\cdot(1+2+4\cdot4) = 96$.

Remark 6.25. *By replacing α with C, as discussed in Remark 6.11, the matrix M constructed from B_1, B_2 and B_3 in (6.17) becomes an NMDS over $GL(8, \mathbb{F}_2)$. Furthermore, the binary matrices C^2 and C^{-2} can be implemented with 2 XORs. Consequently, the implementation cost of the matrix M is 176 XORs over $GL(8, \mathbb{F}_2)$.*

6.5.5 Construction of 8×8 nonrecursive NMDS matrices

In this section, we present a lightweight 8×8 matrix M over the field $\mathbb{F}_{2^4}$ that can be implemented with 108 XORs, which meets the best known result. To construct the matrix M, we use three GDLS matrices, B_1, B_2, and B_3, of order 8, by $M = B_2B_1B_3B_2^3$. These GDLS matrices are generated using the permutations $\rho_1 = [4, 5, 2, 3, 8, 1, 6, 7]$ and $\rho_2 = [5, 4, 3, 6, 1, 8, 7, 2]$, along with the following diagonal matrices.

(i) B_1: $\rho_1, \rho_2, D_1 = diag(1, \alpha, 1, \alpha,\ 1, \alpha, 1, \alpha)$ and $D_2 = diag(1, 0, 1, 0,\ 1, 0, 1, 0)$

(ii) B_2: $\rho_1, \rho_2, D_1 = diag(1, 1, 1, 1,\ 1, 1, 1, 1)$, $D_2 = diag(1, 0, 1, 0,\ 1, 0, 1, 0)$

(iii) B_3: $\rho_1, \rho_2, D_1 = diag(1, 1, 1, 1,\ 1, 1, 1, 1)$ and $D_2 = diag(\alpha^{-2}, 0, \alpha^{-2}, 0,\ \alpha^{-2}, 0, \alpha^{-2}, 0)$

Table 6.6: A summary of results on NMDS matrices of this chapter.

Order n	**Input**	**Type**	**Iterations**	**XOR count**
4	4-bit	recursive	3	8
4	4-bit	nonrecursive	-	24
4	8-bit	recursive	3	16
4	8-bit	nonrecursive	-	48
5	4-bit	recursive	4	14
5	4-bit	recursive	5	13
5	4-bit	nonrecursive	-	50
5	8-bit	recursive	4	26
5	8-bit	recursive	5	25
5	8-bit	nonrecursive	-	98
6	4-bit	recursive	5	14
6	4-bit	recursive	6	13
6	4-bit	nonrecursive	-	65
6	8-bit	recursive	4	26
6	8-bit	recursive	5	25
6	8-bit	nonrecursive	-	125
7	4-bit	recursive	5	22
7	4-bit	recursive	6	20
7	4-bit	recursive	7	18
7	4-bit	nonrecursive	-	96
7	8-bit	recursive	5	38
7	8-bit	recursive	6	36
7	8-bit	recursive	7	34
7	8-bit	nonrecursive	-	176
8	4-bit	nonrecursive	-	108
8	8-bit	recursive	7	40
8	8-bit	recursive	8	38
8	8-bit	nonrecursive	-	204

$$B_1=\begin{bmatrix}0&0&0&0&1&\alpha&0&0\\0&0&1&0&0&0&0&0\\0&0&1&\alpha&0&0&0&0\\1&0&0&0&0&0&0&0\\1&\alpha&0&0&0&0&0&0\\0&0&0&0&0&0&1&0\\0&0&0&0&0&0&1&\alpha\\0&0&0&0&1&0&0&0\end{bmatrix} B_2=\begin{bmatrix}0&0&0&0&1&1&0&0\\0&0&1&0&0&0&0&0\\0&0&1&1&0&0&0&0\\1&0&0&0&0&0&0&0\\1&1&0&0&0&0&0&0\\0&0&0&0&0&0&1&0\\0&0&0&0&0&0&1&1\\0&0&0&0&1&0&0&0\end{bmatrix} B_3=\begin{bmatrix}0&0&0&0&\alpha^{-2}&1&0&0\\0&0&1&0&0&0&0&0\\0&0&\alpha^{-2}&1&0&0&0&0\\1&0&0&0&0&0&0&0\\\alpha^{-2}&1&0&0&0&0&0&0\\0&0&0&0&0&0&1&0\\0&0&0&0&0&0&\alpha^{-2}&1\\0&0&0&0&1&0&0&0\end{bmatrix}, \tag{6.18}$$

where α is a primitive element of $\mathbb{F}_{2^4}$ with $\alpha^4+\alpha+1=0$. Therefore, M can be implemented with $(1+1+1+1+4\cdot 4)+4\cdot(0+4\cdot 4)+(2+2+2+2+4\cdot 4)=108$ XORs.

Remark 6.26. *As discussed in Remark 6.11, if we substitute α with C, the matrix M that is formed from B_1, B_2, and B_3 in (6.18) becomes an NMDS matrix over $GL(8,\mathbb{F}_2)$. Also, the binary matrix C^{-2} can be implemented with only 2 XORs. Therefore, the implementation cost of the matrix M becomes 204 XORs over $GL(8,\mathbb{F}_2)$.*

6.6 Conclusion

In this chapter, we have explored the construction of NMDS matrices using both recursive and nonrecursive approaches. We have presented various theoretical results and introduced lightweight NMDS matrices of different orders in both recursive and nonrecursive approaches. Table 6.6 provides an overview of the implementation cost of the NMDS matrices presented in this chapter. Furthermore, to compare with MDS matrices, we examine some well-known results of MDS matrices and apply them to NMDS matrices. For instance, Table 6.4 compares the involutory and orthogonal properties of MDS and NMDS matrices constructed from circulant, left-circulant, Toeplitz, and Hankel matrices.

7

On the Direct Construction of MDS and Near-MDS Matrices

Contents

7.1 Introduction

The optimal branch number of MDS matrices makes them a preferred choice for designing diffusion layers in many block ciphers and hash functions. Consequently, various methods have been proposed for designing MDS matrices, including search and direct methods. While exhaustive search is suitable for small order MDS matrices, direct constructions are preferred for larger orders due to the vast search space involved. In the literature, there has been extensive research on the direct construction of MDS matrices using both recursive and nonrecursive methods. On the other hand, in lightweight cryptography, NMDS matrices with sub-optimal branch numbers offer a better balance between security and efficiency as a diffusion layer compared to MDS matrices. Despite their potential benefits, research on NMDS matrices has been limited in the literature, and there is currently no direct construction method available for them in a recursive approach.

This chapter aims to bridge the gap by presenting direct constructions of NMDS matrices in the recursive setting. It also introduces a new direct construction technique for recursive MDS matrices. Furthermore, the chapter introduces generalized Vandermonde matrices for direct constructions of nonrecursive MDS and NMDS matrices. Additionally, a method for constructing involutory MDS and NMDS matrices is proposed.

Outline: The rest of this chapter is structured as follows: Section 7.2 presents direct constructions of nonrecursive MDS and NMDS matrices. Section 7.3 focuses on direct constructions of NMDS matrices in recursive approaches, including a new direct construction of recursive MDS matrices. Lastly, Section 7.4 concludes the chapter.

7.2 Direct Construction of Nonrecursive MDS and NMDS Matrices

The application of Vandermonde matrices for constructing MDS codes is well documented in literature [GR13a, LF04a, LF04b, MRS12, SDMO12]. In this section, we explore the use of generalized Vandermonde matrices for the construction of both MDS and NMDS matrices. Specifically, we focus on the generalized Vandermonde matrices $V_\perp(\mathbf{x}; I)$, where I is a subset of $\{1, n-1, n\}$.

Generalized Vandermonde matrices, with these parameters, defined over a finite field can contain singular submatrices (see Example 7.1). Consequently, these matrices by itself need not be MDS over a finite field. However, like Vandermonde based constructions, we can use two generalized Vandermonde matrices for constructing MDS matrices.

Example 7.1. *Consider the generalized Vandermonde matrix* $V_\perp(\mathbf{x}; I)$ *with* $\mathbf{x} = (1, \alpha, \alpha^2, \alpha^5)$ *and* $I = \{3\}$

$$V_\perp(\mathbf{x}; I) = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & \alpha & \alpha^2 & \alpha^5 \\ 1 & \alpha^2 & \alpha^4 & \alpha^{10} \\ 1 & \alpha^4 & \alpha^8 & \alpha^{20} \end{bmatrix},$$

where α *is a primitive element of the finite field* $\mathbb{F}_{2^4}$ *constructed by the polynomial* $x^4 + x + 1$. *Consider the* 2×2 *submatrix*

$$\begin{bmatrix} 1 & \alpha^5 \\ 1 & \alpha^{20} \end{bmatrix}$$

which is singular as $\alpha^{20} = \alpha^5$.

Theorem 7.1. *Let* $V_1 = V_\perp(\mathbf{x}; I)$ *and* $V_2 = V_\perp(\mathbf{y}; I)$ *be two generalized Vandermonde matrices with* $\mathbf{x} = (x_1, x_2, \ldots, x_n)$, $\mathbf{y} = (x_{n+1}, x_{n+2}, \ldots, x_{2n})$ *and* $I = \{n-1\}$. *The elements* x_i *are* $2n$ *distinct elements from* $\mathbb{F}_q$, *and* $\sum_{i=1}^{n} x_{r_i} \neq 0$ *for all* $R = \{r_1, r_2, \ldots, r_n\} \subset E$, *where* $E = \{1, 2, \ldots, 2n\}$. *Then the matrices* $V_1^{-1}V_2$ *and* $V_2^{-1}V_1$ *are such that any square submatrix of them is nonsingular and hence MDS matrices.*

Proof. Let U be the $n \times 2n$ matrix $[V_1 \mid V_2]$. By Corollary 2.10, we can conclude that both V_1 and V_2 are nonsingular matrices. Consider the product $G = V_1^{-1}U = [I \mid A]$, where $A = V_1^{-1}V_2$. We will now prove that A does not contain any singular submatrix.

Now, since $U = V_1 G$, from Lemma 2.5, we can say that U is also a generator matrix for the linear code $\mathcal{C}$ generated by matrix $G = [I \mid A]$. From Remark 2.3, we know that a generator matrix U generates an $[2n, n, n+1]$ MDS code if and only if any n columns of U is linearly independent.

Now we can observe that any n columns of U form a generalized Vandermonde matrix of the same form as V_1 and V_2. Since each x_i are distinct and $\sum_{i=1}^{n} x_{r_i} \neq 0$ for all $R = \{r_1, r_2, \ldots, r_n\} \subset E$, form Corollary 2.10, we can say that every n columns of U are linearly independent. Hence, we can say that the code $\mathcal{C}$ is an MDS code.

Therefore, G generates an $[2n, n, n+1]$ MDS code and hence $A = V_1^{-1}V_2$ is an MDS matrix. For $V_2^{-1}V_1$, the proof is identical. □

Remark 7.1. *We know that the inverse of an MDS matrix is again MDS (See Corollary 2.7), therefore, if* $V_1^{-1}V_2$ *is MDS, then* $V_2^{-1}V_1$ *is also MDS and vice versa.*

Example 7.2. *Consider the generalized Vandermonde matrices* $V_1 = V_\perp(\mathbf{x}; I)$ *and* $V_2 = V_\perp(\mathbf{y}; I)$ *with* $\mathbf{x} = (1, \alpha, \alpha^2, \alpha^3)$, $\mathbf{y} = (\alpha^4, \alpha^5, \alpha^6, \alpha^7)$ *and* $I = \{3\}$, *where* α *is a primitive element of* $\mathbb{F}_{2^8}$ *with* $\alpha^8 + \alpha^7 + \alpha^6 + \alpha + 1 = 0$. *It can be verified that* V_1 *and* V_2 *satisfy the conditions in Theorem 7.1. Therefore, the matrices*

$$V_1^{-1}V_2 = \begin{bmatrix} \alpha^7 & \alpha^{234} & \alpha^{57} & \alpha^{156} \\ \alpha^{37} & \alpha^{66} & \alpha^{55} & \alpha^{211} \\ \alpha^{205} & \alpha^{100} & \alpha^{30} & \alpha^{86} \\ \alpha^{227} & \alpha^{50} & \alpha^{149} & \alpha^{40} \end{bmatrix} \text{ and } V_2^{-1}V_1 = \begin{bmatrix} \alpha^{136} & \alpha^{49} & \alpha^{235} & \alpha^{30} \\ \alpha^{210} & \alpha^{77} & \alpha^{201} & \alpha^{198} \\ \alpha^{144} & \alpha^{72} & \alpha^{52} & \alpha^{220} \\ \alpha^{42} & \alpha^{228} & \alpha^{23} & \alpha^{248} \end{bmatrix}$$

are MDS matrices.

Similar to MDS matrices, generalized Vandermonde matrices with $I = \{n-1\}$ themselves may not be NMDS over a finite field (see Example 7.3). As a consequence, we use two generalized Vandermonde matrices for constructing NMDS matrices.

Example 7.3. *Consider the generalized Vandermonde matrix* $A = V_\perp(\mathbf{x}; I)$ *with* $\mathbf{x} = (1, \alpha, \alpha^3, \alpha^7)$ *and* $I = \{3\}$.

$$A = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & \alpha & \alpha^3 & \alpha^7 \\ 1 & \alpha^2 & \alpha^6 & \alpha^{14} \\ 1 & \alpha^4 & \alpha^{12} & \alpha^{28} \end{bmatrix},$$

where α *is a primitive element of* $\mathbb{F}_{2^4}$ *with* $\alpha^4 + \alpha + 1 = 0$. *Now consider the generator matrix*

$$G = [I \mid A] = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & \alpha & \alpha^3 & \alpha^7 \\ 0 & 0 & 1 & 0 & 1 & \alpha^2 & \alpha^6 & \alpha^{14} \\ 0 & 0 & 0 & 1 & 1 & \alpha^4 & \alpha^{12} & \alpha^{28} \end{bmatrix}.$$

Now consider matrix

$$M = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & \alpha & \alpha^3 & \alpha^7 \\ 1 & 1 & \alpha^2 & \alpha^6 & \alpha^{14} \\ 0 & 1 & \alpha^4 & \alpha^{12} & \alpha^{28} \end{bmatrix},$$

which is constructed by the five columns: the third, fifth, sixth, seventh, and eighth columns of G. *It can be observed that* $\operatorname{rank}(M) = 3 < 4$, *which violates the condition (iii) in Lemma 2.4. Therefore,* A *is not an NMDS matrix.*

Theorem 7.2. *Let* $V_1 = V_\perp(\mathbf{x}; I)$ *and* $V_2 = V_\perp(\mathbf{y}; I)$ *be two generalized Vandermonde matrices with* $\mathbf{x} = (x_1, x_2, \ldots, x_n)$, $\mathbf{y} = (x_{n+1}, x_{n+2}, \ldots, x_{2n})$ *and* $I = \{n-1\}$. *The elements* x_i *are* $2n$ *distinct elements from* $\mathbb{F}_q$ *such that* $\sum_{i=1}^{n} x_i \neq 0$, $\sum_{i=1}^{n} x_{n+i} \neq 0$ *and* $\sum_{i=1}^{n} x_{r_i} = 0$ *for some other* $R = \{r_1, r_2, \ldots, r_n\} \subset E$, *where* $E = \{1, 2, \ldots, 2n\}$. *Then the matrices* $V_1^{-1}V_2$ *and* $V_2^{-1}V_1$ *are NMDS matrices.*

Proof. Let U be the $n \times 2n$ matrix $[V_1 \mid V_2]$. By Corollary 2.10, we can conclude that both V_1 and V_2 are nonsingular matrices. Consider the product $G = V_1^{-1}U = [I \mid A]$, where $A = V_1^{-1}V_2$. To show, $A = V_1^{-1}V_2$ is an NMDS matrix, we need to prove that the $[2n, n]$ linear code $\mathcal{C}$ generated by $G = [I \mid A]$ is an NMDS code.

Now, since $U = V_1 G$, from Lemma 2.5, we can say that U is also a generator matrix for the linear code $\mathcal{C}$. Therefore, we can conclude that $A = V_1^{-1}V_2$ is an NMDS matrix if and only if U meets the three conditions mentioned in Lemma 2.4.

A submatrix $U[R]$, constructed from any t columns of U, is given by

$$U[R] = \begin{bmatrix} 1 & 1 & \dots & 1 \\ x_{r_1} & x_{r_2} & \dots & x_{r_t} \\ x_{r_1}^2 & x_{r_2}^2 & \dots & x_{r_t}^2 \\ \vdots & \vdots & \ddots & \vdots \\ x_{r_1}^{n-2} & x_{r_2}^{n-2} & \dots & x_{r_t}^{n-2} \\ x_{r_1}^n & x_{r_2}^n & \dots & x_{r_t}^n \end{bmatrix}, \tag{7.1}$$

where R denotes a set $\{r_1, r_2, \dots, r_t\} \subset E = \{1, 2, \dots, 2n\}$ of t elements.

So for $R = \{r_1, r_2, \dots, r_{n-1}\} \subset E$ we have

$$U[R] = \begin{bmatrix} 1 & 1 & \dots & 1 \\ x_{r_1} & x_{r_2} & \dots & x_{r_{n-1}} \\ \vdots & \vdots & \ddots & \vdots \\ x_{r_1}^{n-2} & x_{r_2}^{n-2} & \dots & x_{r_{n-1}}^{n-2} \\ x_{r_1}^n & x_{r_2}^n & \dots & x_{r_{n-1}}^n \end{bmatrix}.$$

Now, we consider the $(n-1) \times (n-1)$ submatrix $U'[R]$ of $U[R]$, which is constructed from the first $n-1$ rows of $U[R]$. Therefore, we have

$$\begin{aligned} U'[R] &= \begin{bmatrix} 1 & 1 & \dots & 1 \\ x_{r_1} & x_{r_2} & \dots & x_{r_{n-1}} \\ \vdots & \vdots & \ddots & \vdots \\ x_{r_1}^{n-3} & x_{r_2}^{n-3} & \dots & x_{r_{n-1}}^{n-3} \\ x_{r_1}^{n-2} & x_{r_2}^{n-2} & \dots & x_{r_{n-1}}^{n-2} \end{bmatrix} \\ &= vand(x_{r_1}, x_{r_2}, \dots, x_{r_{n-1}}), \end{aligned}$$

which is nonsingular since each x_i is a distinct element. Therefore, any submatrix of U constructed from any $n-1$ columns has a nonsingular $(n-1) \times (n-1)$ submatrix, implying that any $n-1$ columns of U are linearly independent.

Now suppose $\sum_{i=1}^{n} x_{r'_i} = 0$ for some $R' = \{r'_1, r'_2, \dots, r'_n\} \subset E$. Then for R', we have

$$U[R'] = \begin{bmatrix} 1 & 1 & \dots & 1 \\ x_{r'_1} & x_{r'_2} & \dots & x_{r'_n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{r'_1}^{n-2} & x_{r'_2}^{n-2} & \dots & x_{r_{n'}}^{n-2} \\ x_{r'_1}^n & x_{r'_2}^n & \dots & x_{r'_n}^n \end{bmatrix},$$

which is a generalized Vandermonde matrix $V_\perp(\mathbf{x}; I)$ with $\mathbf{x} = (x_{r'_1}, x_{r'_2}, \ldots, x_{r'_n})$ and $I = \{n-1\}$. Thus, from Corollary 2.10, we have

$$\det(U[R']) = \left[\prod_{1\le i<j\le n} (x_{r'_j} - x_{r'_i})\right]\left(\sum_{i=1}^{n} x'_{r_i}\right).$$

Since $\sum_{i=1}^{n} x_{r'_i} = 0$, we have $\det(U[R']) = 0$ i.e. the columns of $U[R']$ are linearly dependent. Hence, there exist n columns (depends upon R') that are linearly dependent.

Now we need to show that the third condition of Lemma 2.4 is also satisfied by U. To prove this, we will use a contradiction argument. Suppose, for the sake of contradiction, that each set of $n+1$ columns of U is not of full rank. Let $R'' = \{r_1, r_2, \ldots, r_n, r_{n+1}\} \subset E$ be a set of $n+1$ elements such that the corresponding submatrix $U[R'']$ of U is not of full rank i.e., $\text{rank}(U[R'']) < n$. Now by our assumption, each $n \times n$ submatrix of $U[R'']$ is singular. Since each $x_r \neq x_{r'}$ for $r, r' \in E$, from Corollary 2.10, it follows that

$$\begin{aligned}
x_{r_2} + x_{r_3} + x_{r_4} + x_{r_5} + \cdots + x_{r_{n+1}} &= 0\\
x_{r_1} + x_{r_3} + x_{r_4} + x_{r_5} + \cdots + x_{r_{n+1}} &= 0\\
x_{r_1} + x_{r_2} + x_{r_3} + x_{r_5} + \cdots + x_{r_{n+1}} &= 0\\
&\vdots\\
x_{r_1} + x_{r_2} + x_{r_3} + x_{r_4} + \cdots + x_{r_n} &= 0.
\end{aligned}$$

This system of equations can be written as $MX = 0$, where M is a $(n+1)\times(n+1)$ matrix given by

$$M = \begin{bmatrix} 0 & 1 & 1 & 1 & \ldots 1\\ 1 & 0 & 1 & 1 & \ldots 1\\ 1 & 1 & 0 & 1 & \ldots 1\\ \vdots & \vdots & \vdots & \vdots & \ddots \vdots\\ 1 & 1 & 1 & 1 & \ldots 0 \end{bmatrix} \quad \text{and} \quad X = [x_{r_1}, x_{r_2}, x_{r_3}, \ldots, x_{r_{n+1}}]^T.$$

Note that $\det(M)=(-1)^n n$. Suppose p is the characteristic of the field $\mathbb{F}_q$. We will now examine two scenarios: first, when p does not divide n; and second, when p divides n.

Case 1: $p \nmid n$.

In this case, we have $\det(M) \neq 0$. Therefore, $MX = 0$ has a unique solution $X =$

$[0, 0, \ldots, 0]^T$. This means $x_{r_i} = 0$ for $i = 1, 2, \ldots, n+1$ which is a contradiction because each x_i is distinct.

Case 2: $p|n$.
If $p|n$, M is a singular matrix. Consider the $n \times n$ submatrix M' obtained by excluding the first row and first column of M. The determinant of M' is given by $\det(M') = (-1)^{n-1}(n-1)$. Since p is a prime and $p|n$, we must have $p \nmid (n-1)$. Therefore, $\det(M') \neq 0$. From this, we conclude that the rank of M is n and so the solution space of $MX = 0$ has dimension 1.

Since $p|n$, it is easy to verify that $[1, 1, \ldots, 1]^T$ is a solution of $MX = 0$. As this vector is nonzero, we deduce that the solution space of $MX = 0$ is given by

$$X = \left\{ c \cdot [1, 1, \ldots, 1]^T : \ c \in \mathbb{F}_q \right\}.$$

Therefore, we have

$$[x_{r_1}, x_{r_2}, x_{r_3}, \ldots, x_{r_{n+1}}]^T = c \cdot [1, 1, \ldots, 1]^T$$

for some $c \in \mathbb{F}_q$, which contradicts the fact that each $x_r \neq x_{r'}$ for $r, r' \in E$.

Thus, we can conclude that U, and hence $G = [I \mid A]$, generates an $[2n, n]$ linear NMDS code. Therefore, according to Definition 2.11, $A = V_1^{-1}V_2$ is an NMDS matrix. For $V_2^{-1}V_1$, the proof is identical. □

Remark 7.2. *In Theorem 7.2, it is assumed that* $\sum_{i=1}^{n} x_i \neq 0$ *and* $\sum_{i=1}^{n} x_{n+i} \neq 0$. *This assumption is made based on Corollary 2.10, which states that* $\det(V_\perp(\mathbf{x}; I)) = \det(vand(\mathbf{x}))(\sum_{i=1}^{n} x_i)$ *and* $\det(V_\perp(\mathbf{y}; I)) = \det(vand(\mathbf{y}))(\sum_{i=1}^{n} x_{n+i})$. *If either of these sums is zero, it would result in the determinant of either* V_1 *or* V_2 *being zero, making them singular. Hence, the assumption is necessary to ensure the nonsingularity of* V_1 *and* V_2.

Example 7.4. *Consider the generalized Vandermonde matrices* $V_1 = V_\perp(\mathbf{x}; I)$ *and* $V_2 = V_\perp(\mathbf{y}; I)$ *with* $\mathbf{x} = (1, \alpha, \alpha^2, \alpha^3)$, $\mathbf{y} = (\alpha^4, \alpha^5, \alpha^6, \alpha^7)$ *and* $I = \{3\}$, *where* α *is a primitive element of* $\mathbb{F}_{2^4}$ *with* $\alpha^4 + \alpha + 1 = 0$. *It is easy to check that each* x_i *are distinct and* $1 + \alpha + \alpha^3 + \alpha^7 = 0$. *Therefore, the matrices*

$$V_1^{-1}V_2 = \begin{bmatrix} \alpha^7 & \alpha^9 & \alpha^9 & 1 \\ \alpha^{14} & \alpha^{14} & \alpha^3 & 1 \\ \alpha^{10} & \alpha^5 & \alpha^5 & 0 \\ \alpha^2 & \alpha^2 & \alpha^8 & 1 \end{bmatrix} \text{ and } V_2^{-1}V_1 = \begin{bmatrix} 0 & \alpha^7 & 1 & \alpha^7 \\ 1 & \alpha^{14} & 0 & \alpha^3 \\ 1 & \alpha^5 & 1 & \alpha^{10} \\ 1 & \alpha^8 & 1 & \alpha^8 \end{bmatrix}$$

are NMDS matrices.

Remark 7.3. *Based on the conditions of Theorem 7.2, both V_1 and V_2 are nonsingular matrices. As both V_1 and V_2 are nonsingular, from Lemma 2.9, we can conclude that if $V_1^{-1}V_2$ is an NMDS matrix, then $V_2^{-1}V_1$ is also an NMDS matrix, and vice versa.*

In the context of implementing block ciphers, we know that if an efficient matrix M used in encryption is involutory, then its inverse $M^{-1} = M$ applied for decryption will also be efficient. Hence, it is important to find MDS or NMDS matrices that are also involutory. In the following theorem, we prove a result for obtaining involutory matrices from the generalized Vandermonde matrices with $I = \{n-1\}$. The proof technique used in this theorem follows a similar approach to the proof of the Theorem 3.5 for Vandermonde matrices.

Theorem 7.3. *Let $V_1 = V_\perp(\mathbf{x}; I)$ and $V_2 = V_\perp(\mathbf{y}; I)$ be two generalized Vandermonde matrices of even order over $\mathbb{F}_{2^r}$ with $\mathbf{x} = (x_1, x_2, \ldots, x_n)$, $\mathbf{y} = (y_1, y_2, \ldots, y_n)$ and $I = \{n-1\}$. If $y_i = l + x_i$ for $i = 1, 2, \ldots, n$, for some $l \in \mathbb{F}_{2^r}^\star$ then $V_2{V_1}^{-1}$ is a lower triangular matrix whose nonzero elements are determined by powers of l. Also, $V_1^{-1}V_2$ $(= V_2^{-1}V_1)$ is an involutory matrix.*

Proof. Let $V_1^{-1} = (t_{i,j})_{n,n}$ and $V = V_2V_1^{-1} = (v_{i,j})_{n,n}$. As $V_1V_1^{-1} = I$, we have

$$V_{1_{row(1)}} \cdot V^{-1}_{1_{column(1)}} = \sum_{i=1}^{n} t_{i,1} = 1 \tag{7.2}$$

$$V_{1_{row(k)}} \cdot V^{-1}_{1_{column(1)}} = \sum_{i=1}^{n} x_i^{k-1} \cdot t_{i,1} = 0 \text{ for } 2 \le k \le n-1 \text{ and} \tag{7.3}$$

$$V_{1_{row(n)}} \cdot V^{-1}_{1_{column(1)}} = \sum_{i=1}^{n} x_i^{n} \cdot t_{i,1} = 0. \tag{7.4}$$

Therefore, from Equation 7.2, we have $v_{1,1} = V_{2_{row(1)}} \cdot V^{-1}_{1_{column(1)}} = 1$.

Now for $2 \le k \le n-1$, we have

$$\begin{aligned} v_{k,1} &= V_{2_{row(k)}} \cdot V^{-1}_{1_{column(1)}} \\ &= \sum_{i=1}^{n} y_i^{k-1} \cdot t_{i,1} = \sum_{i=1}^{n} (l + x_i)^{k-1} \cdot t_{i,1} \\ &= \sum_{i=1}^{n} ({}^{k-1}C_0 x_i^{k-1} + {}^{k-1}C_1 x_i^{k-2} \cdot l + \ldots \\ &\qquad + {}^{k-1}C_{k-2} x_i \cdot l^{k-2} + {}^{k-1}C_{k-1} l^{k-1}) \cdot t_{i,1} \\ &= \sum_{i=1}^{n} l^{k-1} \cdot t_{i,1} = l^{k-1} \qquad \text{[By Equation 7.3]}. \end{aligned}$$

Also, we have

$$\begin{aligned}
v_{n,1} &= V_{2_{row(n)}} \cdot V^{-1}_{1_{column(1)}} \\
&= \sum_{i=1}^{n} y_i^n \cdot t_{i,1} = \sum_{i=1}^{n} (l + x_i)^n \cdot t_{i,1} \\
&= \sum_{i=1}^{n} ({}^nC_0 x_i^n + {}^nC_1 x_i^{n-1} \cdot l + \ldots + {}^nC_{n-1} x_i \cdot l^{n-1} + {}^nC_n l^n) \cdot t_{i,1} \\
&= \sum_{i=1}^{n} {}^nC_1 x_i^{n-1} l \cdot t_{i,1} + \sum_{i=1}^{n} l^n \cdot t_{i,1} \qquad \text{[By Equations 7.3 and 7.4]} \\
&= l^n \qquad \text{[Since } n \text{ is even, } {}^nC_1 = 0 \text{ in } \mathbb{F}_{2^r} \text{ and by Equation 7.2].}
\end{aligned}$$

So we have computed the 1st column of $V = V_2 V_1^{-1}$.

Again since $V_1 V_1^{-1} = I$, we have

$$V_{1_{row(1)}} \cdot V^{-1}_{1_{column(2)}} = \sum_{i=1}^{n} t_{i,2} = 0, l \tag{7.5}$$

$$V_{1_{row(2)}} \cdot V^{-1}_{1_{column(2)}} = \sum_{i=1}^{n} x_i \cdot t_{i,2} = 1, \tag{7.6}$$

$$V_{1_{row(k)}} \cdot V^{-1}_{1_{column(2)}} = \sum_{i=1}^{n} x_i^{k-1} \cdot t_{i,2} = 0 \text{ for } 3 \leq k \leq n-1 \text{ and} \tag{7.7}$$

$$V_{1_{row(n)}} \cdot V^{-1}_{1_{column(2)}} = \sum_{i=1}^{n} x_i^{n} \cdot t_{i,2} = 0. \tag{7.8}$$

Therefore, from Equation 7.5, we have $v_{1,2} = V_{2_{row(1)}} \cdot V^{-1}_{1_{column(2)}} = 0$.

Also, we have

$$\begin{aligned}
v_{2,2} &= V_{2_{row(2)}} \cdot V^{-1}_{1_{column(2)}} = \sum_{i=1}^{n} y_i \cdot t_{i,2} \\
&= \sum_{i=1}^{n} (l + x_i) \cdot t_{i,2} = \sum_{i=1}^{n} l \cdot t_{i,2} + \sum_{i=1}^{n} x_i \cdot t_{i,2} = 1 \qquad \text{[By Equations 7.5 and 7.6]}
\end{aligned}$$

Now for $3 \leq k \leq n-1$, we have

$$\begin{aligned}
v_{k,2} &= V_{2_{row(k)}} \cdot V^{-1}_{1_{column(2)}} \\
&= \sum_{i=1}^{n} y_i^{k-1} \cdot t_{i,2} = \sum_{i=1}^{n} (l + x_i)^{k-1} \cdot t_{i,2} \\
&= \sum_{i=1}^{n} ({}^{k-1}C_0 x_i^{k-1} + {}^{k-1}C_1 x_i^{k-2} \cdot l + \ldots \\
&\qquad + {}^{k-1}C_{k-2} x_i \cdot l^{k-2} + {}^{k-1}C_{k-1} l^{k-1}) \cdot t_{i,2}
\end{aligned}$$

$$= \sum_{i=1}^{n} {}^{k-1}C_{k-2} x_i l^{k-2} \cdot t_{i,2} + \sum_{i=1}^{n} l^{k-1} \cdot t_{i,2} \qquad \text{[By Equation 7.7]}$$

$$= {}^{k-1}C_1 l^{k-2} \qquad \text{[By Equations 7.5 and 7.6 and since } {}^{k-1}C_{k-2} = {}^{k-1}C_1\text{]}.$$

Also, we have

$$v_{n,2} = V_{2_{row(n)}} \cdot V^{-1}_{1_{column(2)}}$$

$$= \sum_{i=1}^{n} y_i^n \cdot t_{i,2} = \sum_{i=1}^{n} (l + x_i)^n \cdot t_{i,2}$$

$$= \sum_{i=1}^{n} ({}^nC_0 x_i^n + {}^nC_1 x_i^{n-1} \cdot l + \ldots + {}^nC_{n-1} x_i \cdot l^{n-1} + {}^nC_n l^n) \cdot t_{i,2}$$

$$= \sum_{i=1}^{n} {}^nC_1 x_i^{n-1} l \cdot t_{i,2} + \sum_{i=1}^{n} {}^nC_{n-1} x_i l^{n-1} \cdot t_{i,2} \quad \text{[By Equations 7.5, 7.7 and 7.8]}$$

$$= {}^nC_1 l^{n-1} = 0 \qquad \text{[Since } n \text{ is even, } {}^nC_1 = 0 \text{ in } \mathbb{F}_{2^r} \text{ and by Equation 7.6]}.$$

So we have computed the 2nd column of $V = V_2 V_1^{-1}$. Similarly,

$v_{1,3} = v_{2,3} = 0, v_{3,3} = 1, v_{k,3} = {}^{k-1}C_2 l^{k-3}$ for $4 \leq k \leq n-1$ and

$v_{n,3} = {}^nC_2 l^{n-2}$

$v_{1,4} = v_{2,4} = v_{3,4} = 0, v_{4,4} = 1, v_{k,4} = {}^{k-1}C_3 l^{k-4}$ for $5 \leq k \leq n-1$ and

$v_{n,4} = {}^nC_3 l^{n-3}$ and so on.

Therefore, $V = V_2 V_1^{-1}$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 & \ldots\ldots & 0 & 0 \\ l & 1 & 0 & 0 & \ldots\ldots & 0 & 0 \\ l^2 & {}^2C_1 l & 1 & 0 & \ldots\ldots & 0 & 0 \\ l^3 & {}^3C_1 l^2 & {}^3C_2 l & 1 & \ldots\ldots & 0 & 0 \\ l^4 & {}^4C_1 l^3 & {}^4C_2 l^2 & {}^4C_3 l & \ldots\ldots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ldots\ldots & \vdots & \vdots \\ l^{n-2} & {}^{n-2}C_1 l^{n-3} & {}^{n-2}C_2 l^{n-4} & {}^{n-2}C_3 l^{n-5} & \ldots\ldots & 1 & 0 \\ l^n & {}^nC_1 l^{n-1} & {}^nC_2 l^{n-2} & {}^nC_3 l^{n-3} & \ldots\ldots & {}^nC_{n-2} l^2 & 1 \end{bmatrix}.$$

Thus, $V_2 V_1^{-1}$ is a lower triangular matrix. Therefore, for $1 \leq i \leq n-1$ and $1 \leq j \leq n$, we have

$$(VV_2)_{i,j} = V_{row(i)} \cdot V_{2_{column(j)}}$$

$$= l^{i-1} + {}^{i-1}C_1 l^{i-2} \cdot y_j + {}^{i-1}C_2 l^{i-3} \cdot y_j^2 + \ldots + {}^{i-1}C_{i-2} l \cdot y_j^{i-2} + y_j^{i-1}$$

$$= (l + y_j)^{i-1} = x_j^{i-1} = (V_1)_{i,j}.$$

Now for $1 \leq j \leq n$, we have

$$\begin{aligned}(VV_2)_{n,j} &= V_{row(n)} \cdot V_{2_{column(j)}} \\ &= l^n + {}^nC_1 l^{n-1} \cdot y_j + {}^nC_2 l^{n-2} \cdot y_j^2 + \ldots + {}^nC_{n-2} l^2 \cdot y_j^{n-2} + y_j^n \\ &= l^n + {}^nC_1 l^{n-1} \cdot y_j + {}^nC_2 l^{n-2} \cdot y_j^2 + \ldots + {}^nC_{n-2} l^2 \cdot y_j^{n-2} \\ &\qquad + {}^nC_{n-1} l \cdot y_j^{n-1} + y_j^n \quad [\text{Since } {}^nC_{n-1} = 0 \text{ in } \mathbb{F}_{2^r}] \\ &= (l + y_j)^n = x_j^n = (V_1)_{n,j}.\end{aligned}$$

Thus, we have $V_2 V_1^{-1} V_2 = V_1$ which implies that $(V_1^{-1} V_2)^2 = I$ i.e. $V_1^{-1} V_2 = V_2^{-1} V_1$ is involutory. □

Now by applying Theorem 7.1 and Theorem 7.3, we can find involutory MDS matrices over $\mathbb{F}_{2^r}$, as follows.

Corollary 7.1. *Let $V_1 = V_\perp(\mathbf{x}; I)$ and $V_2 = V_\perp(\mathbf{y}; I)$ be two generalized Vandermonde matrices of even order over $\mathbb{F}_{2^r}$ with $\mathbf{x} = (x_1, x_2, \ldots, x_n)$, $\mathbf{y} = (x_{n+1}, x_{n+2}, \ldots, x_{2n})$ and $I = \{n-1\}$. If V_1 and V_2 satisfying the three properties:*

(i) $x_{n+i} = l + x_i$ for $i = 1, 2, \ldots, n$, for some $l \in \mathbb{F}_{2^r}^\star$,

(ii) $x_i \neq x_j$ for $i \neq j$ where $1 \leq i, j \leq 2n$, and

(iii) $\sum_{i=1}^{n} x_{r_i} \neq 0$ for all $R = \{r_1, r_2, \ldots, r_n\} \subset E$, where $E = \{1, 2, \ldots, 2n\}$,

then $V_1^{-1} V_2$ is an involutory MDS matrix.

Example 7.5. *Let α be a primitive element of $\mathbb{F}_{2^8}$ with $\alpha^8 + \alpha^7 + \alpha^6 + \alpha + 1 = 0$. Let $l = \alpha$, $\mathbf{x} = (1, \alpha, \alpha^2, \alpha^3, \alpha^4, \alpha^5)$, and $\mathbf{y} = (\alpha + 1, 0, \alpha^2 + \alpha, \alpha^3 + \alpha, \alpha^4 + \alpha, \alpha^5 + \alpha)$. Consider the generalized Vandermonde matrices $V_1 = V_\perp(\mathbf{x}; I)$ and $V_2 = V_\perp(\mathbf{y}; I)$ with $I = \{5\}$. Then it can be checked that both matrices V_1 and V_2 satisfy the conditions of Corollary 7.1. Therefore, the matrix*

$$V_1^{-1} V_2 = \begin{bmatrix} \alpha^{113} & \alpha^{33} & \alpha^{227} & \alpha^{93} & \alpha^{16} & \alpha^{174} \\ \alpha^{63} & \alpha^{107} & \alpha^{186} & \alpha^{149} & \alpha^{175} & \alpha^{10} \\ \alpha^{105} & \alpha^{34} & \alpha^{116} & \alpha^{97} & \alpha^{198} & \alpha^{197} \\ \alpha^{40} & \alpha^{66} & \alpha^{166} & \alpha^{43} & \alpha^{213} & \alpha^{52} \\ \alpha^{136} & \alpha^{10} & \alpha^{185} & \alpha^{131} & \alpha^{5} & \alpha^{136} \\ \alpha^{211} & \alpha^{17} & \alpha^{101} & \alpha^{142} & \alpha^{53} & \alpha^{56} \end{bmatrix}$$

is an involutory MDS matrix.

Remark 7.4. *It is worth mentioning that the above result is not true for odd order matrices. For example, consider the* 3×3 *generalized Vandermonde matrices* $V_1 = V_\perp(\mathbf{x}; I)$ *and* $V_2 = V_\perp(\mathbf{y}; I)$ *with* $I = \{2\}$, $\mathbf{x} = (1, \alpha, \alpha^2)$ *and* $\mathbf{y} = (1+\alpha^3, \alpha+\alpha^3, \alpha^2+\alpha^3)$, *where* α *is a primitive element of* $\mathbb{F}_{2^4}$ *with* $\alpha^4 + \alpha + 1 = 0$. *Then it can be checked that the matrices* V_1 *and* V_2 *satisfy the conditions in Corollary 7.1. However, the matrix*

$$V_1^{-1}V_2 = \begin{bmatrix} \alpha^{10} & \alpha^{13} & \alpha^{1} \\ \alpha^{3} & \alpha^{11} & \alpha^{11} \\ \alpha^{11} & \alpha^{1} & \alpha^{13} \end{bmatrix}$$

is not an involutory matrix.

Also, by using Theorem 7.2 and Theorem 7.3, we can obtain involutory NMDS matrices over $\mathbb{F}_{2^r}$ with the following approach.

Corollary 7.2. *Let* $V_1 = V_\perp(\mathbf{x}; I)$ *and* $V_2 = V_\perp(\mathbf{y}; I)$ *be two generalized Vandermonde matrices of even order over* $\mathbb{F}_{2^r}$ *with* $\mathbf{x} = (x_1, x_2, \ldots, x_n)$, $\mathbf{y} = (x_{n+1}, x_{n+2}, \ldots, x_{2n})$ *and* $I = \{n-1\}$. *If* V_1 *and* V_2 *satisfying the three properties:*

(i) $x_{n+i} = l + x_i$ *for* $i = 1, 2, \ldots, n$, *for some* $l \in \mathbb{F}_{2^r}^\star$,

(ii) $x_i \neq x_j$ *for* $i \neq j$ *where* $1 \leq i, j \leq 2n$, *and*

(iii) $\sum_{i=1}^{n} x_i \neq 0$, $\sum_{i=1}^{n} x_{n+i} \neq 0$ *and* $\sum_{i=1}^{n} x_{r_i} = 0$ *for some other* $R = \{r_1, r_2, \ldots, r_n\} \subset E$, *where* $E = \{1, 2, \ldots, 2n\}$,

then $V_1^{-1}V_2$ *is an involutory NMDS matrix.*

Example 7.6. *Let* $l = 1$, $\mathbf{x} = (1, \alpha, \alpha^2, \alpha^3)$, *and* $\mathbf{y} = (0, 1+\alpha, 1+\alpha^2, 1+\alpha^3)$, *where* α *is a primitive element of* $\mathbb{F}_{2^4}$ *with* $\alpha^4 + \alpha + 1 = 0$. *Consider the generalized Vandermonde matrices* $V_1 = V_\perp(\mathbf{x}; I)$ *and* $V_2 = V_\perp(\mathbf{y}; I)$ *with* $I = \{3\}$. *Then it can be checked that both matrices* V_1 *and* V_2 *satisfy the conditions of Corollary 7.2. Therefore, the matrix*

$$V_1^{-1}V_2 = \begin{bmatrix} \alpha^{9} & \alpha^{7} & \alpha^{7} & \alpha^{7} \\ \alpha^{3} & \alpha^{14} & \alpha^{3} & \alpha^{3} \\ \alpha^{10} & \alpha^{10} & \alpha^{5} & \alpha^{10} \\ \alpha^{2} & \alpha^{2} & \alpha^{2} & \alpha^{8} \end{bmatrix}$$

is an involutory NMDS matrix.

We will now focus on using the generalized Vandermonde matrices $V_\perp(\mathbf{x}; I)$ with $I = \{1\}$ for constructing MDS and NMDS matrices. Similar to the case of generalized Vandermonde matrices with $I = \{n-1\}$, these matrices alone may not be MDS or NMDS (as shown in Example 7.7). Therefore, we will consider two generalized Vandermonde matrices for the construction of MDS and NMDS matrices.

Example 7.7. *Consider the generalized Vandermonde matrix $V_\perp(\mathbf{x}; I)$ with $\mathbf{x} = (1, \alpha, \alpha^5, \alpha^{10})$ and $I = \{1\}$*

$$V_\perp(\mathbf{x}; I) = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & \alpha^2 & \alpha^{10} & \alpha^{20} \\ 1 & \alpha^3 & \alpha^{15} & \alpha^{30} \\ 1 & \alpha^4 & \alpha^{20} & \alpha^{40} \end{bmatrix},$$

where α is a primitive element of $\mathbb{F}_{2^4}$ with $\alpha^4 + \alpha + 1 = 0$. But it contains a singular 2×2 submatrix $\begin{bmatrix} 1 & 1 \\ \alpha^{15} & \alpha^{30} \end{bmatrix}$. Hence, $V_\perp(\mathbf{x}; I)$ is not an MDS matrix. Also, it can be checked that $V_\perp(\mathbf{x}; I)$ is not an NMDS matrix.

By utilizing Corollary 2.11, we can establish the following theorem, resembling the proof technique employed in Theorem 7.1. In the interest of conciseness, we state the result without providing a proof.

Theorem 7.4. *Let $V_1 = V_\perp(\mathbf{x}; I)$ and $V_2 = V_\perp(\mathbf{y}; I)$ be two generalized Vandermonde matrices with $\mathbf{x} = (x_1, x_2, \ldots, x_n)$, $\mathbf{y} = (x_{n+1}, x_{n+2}, \ldots, x_{2n})$ and $I = \{1\}$. Suppose that the elements x_i are $2n$ distinct nonzero elements from $\mathbb{F}_q$, and $\sum_{i=1}^{n} x_{r_i}^{-1} \neq 0$ for all $R = \{r_1, r_2, \ldots, r_n\} \subset E$, where $E = \{1, 2, \ldots, 2n\}$. Then the matrices $V_1^{-1}V_2$ and $V_2^{-1}V_1$ are such that any square submatrix of them is nonsingular and hence MDS matrices.*

Example 7.8. *Consider the generalized Vandermonde matrices $V_1 = V_\perp(\mathbf{x}; I)$ and $V_2 = V_\perp(\mathbf{y}; I)$ with $\mathbf{x} = (1, \alpha, \alpha^2, \alpha^3)$, $\mathbf{y} = (\alpha^4, \alpha^5, \alpha^6, \alpha^7)$ and $I = \{1\}$, where α is a primitive element of $\mathbb{F}_{2^8}$ with $\alpha^8 + \alpha^7 + \alpha^6 + \alpha + 1 = 0$. It can be verified that V_1 and V_2 satisfy the conditions in Theorem 7.4. Therefore, the matrices*

$$V_1^{-1}V_2 = \begin{bmatrix} \alpha^9 & \alpha^{43} & \alpha^{252} & \alpha^{70} \\ \alpha^{232} & \alpha^{68} & \alpha^{92} & \alpha^{168} \\ \alpha^{206} & \alpha^{213} & \alpha^{93} & \alpha^{230} \\ \alpha^{34} & \alpha^{243} & \alpha^{61} & \alpha^{152} \end{bmatrix} \text{ and } V_2^{-1}V_1 = \begin{bmatrix} \alpha^{24} & \alpha^{137} & \alpha^{42} & \alpha^{223} \\ \alpha^{66} & \alpha^{14} & \alpha^{88} & \alpha^{197} \\ \alpha^{187} & \alpha^{35} & \alpha^{50} & \alpha^{25} \\ \alpha^{128} & \alpha^{33} & \alpha^{214} & \alpha^{246} \end{bmatrix}$$

are MDS matrices.

In the following theorem we discuss a new construction of NMDS matrices from the generalized Vandermonde matrices with $I = \{1\}$. The proof can be derived using Corollary 2.11, following a similar approach to that of Theorem 7.2. We state the result without providing a proof.

Theorem 7.5. *Let $V_1 = V_\perp(\mathbf{x}; I)$ and $V_2 = V_\perp(\mathbf{y}; I)$ be two generalized Vandermonde matrices with $\mathbf{x} = (x_1, x_2, \ldots, x_n)$, $\mathbf{y} = (x_{n+1}, x_{n+2}, \ldots, x_{2n})$ and $I = \{1\}$. Assume that the elements x_i are $2n$ distinct nonzero elements from $\mathbb{F}_q$ such that $\sum_{i=1}^{n} x_i^{-1} \neq 0$, $\sum_{i=1}^{n} x_{n+i}^{-1} \neq 0$ and $\sum_{i=1}^{n} x_{r_i}^{-1} = 0$ for some other $R = \{r_1, r_2, \ldots, r_n\} \subset E$, where $E = \{1, 2, \ldots, 2n\}$. Then the matrices $V_1^{-1}V_2$ and $V_2^{-1}V_1$ are NMDS matrices.*

Remark 7.5. *Similar to Theorem 7.2, according to the Corollary 2.11, the assumption $\sum_{i=1}^{n} x_i^{-1} \neq 0$ and $\sum_{i=1}^{n} x_{n+i}^{-1} \neq 0$ in Theorem 7.5 is necessary to ensure the nonsingularity of V_1 and V_2.*

Example 7.9. *Consider the generalized Vandermonde matrices $V_1 = V_\perp(\mathbf{x}; I)$ and $V_2 = V_\perp(\mathbf{y}; I)$ with $\mathbf{x} = (1, \alpha, \alpha^2, \alpha^3)$, $\mathbf{y} = (\alpha^4, \alpha^5, \alpha^6, \alpha^7)$ and $I = \{1\}$, where α is a primitive element of $\mathbb{F}_{2^4}$ with $\alpha^4 + \alpha + 1 = 0$. It is easy to check that each x_i are distinct and $1 + \alpha^{-1} + \alpha^{-2} + \alpha^{-7} = 0$. Therefore, the matrices*

$$V_1^{-1}V_2 = \begin{bmatrix} \alpha^9 & \alpha^5 & \alpha^2 & \alpha^{13} \\ \alpha^7 & \alpha & \alpha^{10} & \alpha^9 \\ \alpha^{11} & 0 & 1 & \alpha^5 \\ \alpha^{11} & \alpha^8 & \alpha^4 & 0 \end{bmatrix} \text{ and } V_2^{-1}V_1 = \begin{bmatrix} \alpha^{14} & \alpha^{11} & \alpha^9 & \alpha^{13} \\ 0 & \alpha^4 & \alpha^8 & \alpha^2 \\ \alpha^6 & \alpha^{13} & \alpha^{13} & \alpha^2 \\ \alpha^2 & 1 & \alpha^4 & \alpha^6 \end{bmatrix}$$

are NMDS matrices.

Now we consider generalized Vandermonde matrices $V(\mathbf{x}; T)$, where T has more than one discontinuity, specifically, we consider $V_\perp(\mathbf{x}; I)$ with $I = \{1, n\}$ for providing a new direct construction for MDS matrices. The proof follows a similar approach to that of Theorem 7.1 and can be derived using Corollary 2.12. For brevity, we state the result without presenting a proof.

Theorem 7.6. *Let $V_1 = V_\perp(\mathbf{x}; I)$ and $V_2 = V_\perp(\mathbf{y}; I)$ be two generalized Vandermonde matrices with $\mathbf{x} = (x_1, x_2, \ldots, x_n)$, $\mathbf{y} = (x_{n+1}, x_{n+2}, \ldots, x_{2n})$ and $I = \{1, n\}$. The elements x_i are $2n$ distinct nonzero elements from $\mathbb{F}_q$, and $(\sum_{i=1}^{n} x_{r_i})(\sum_{i=1}^{n} x_{r_i}^{-1}) - 1 \neq 0$ for all $R = \{r_1, r_2, \ldots, r_n\} \subset E$, where $E = \{1, 2, \ldots, 2n\}$. Then the matrices $V_1^{-1}V_2$ and $V_2^{-1}V_1$ are such that any square submatrix of them is nonsingular and hence MDS matrices.*

Example 7.10. *Consider the generalized Vandermonde matrices* $V_1 = V_\perp(\mathbf{x}; I)$ *and* $V_2 = V_\perp(\mathbf{y}; I)$ *with* $\mathbf{x} = (1, \alpha, \alpha^2, \alpha^3)$, $\mathbf{y} = (\alpha^4, \alpha^5, \alpha^6, \alpha^7)$ *and* $I = \{1, 4\}$, *where* α *is a primitive element of* $\mathbb{F}_{2^4}$ *with* $\alpha^4 + \alpha + 1 = 0$. *It can be verified that* V_1 *and* V_2 *satisfy the conditions in Theorem 7.6. Therefore, the matrices*

$$V_1^{-1}V_2 = \begin{bmatrix} \alpha^{10} & \alpha^2 & \alpha^2 & \alpha^{14} \\ \alpha^{12} & \alpha^2 & \alpha^{10} & \alpha^5 \\ \alpha & \alpha^9 & 1 & 1 \\ \alpha^7 & \alpha^7 & \alpha^4 & \alpha^{12} \end{bmatrix} \text{ and } V_2^{-1}V_1 = \begin{bmatrix} \alpha^7 & \alpha^4 & \alpha^{12} & \alpha^2 \\ \alpha^5 & \alpha^{10} & \alpha^9 & \alpha^6 \\ \alpha^5 & 1 & \alpha^{12} & \alpha^{12} \\ \alpha^9 & \alpha^2 & \alpha^7 & \alpha^5 \end{bmatrix}$$

are MDS matrices.

Remark 7.6. *It is important to note that in Theorem 7.1 and Theorem 7.2, at most one* x_i *may be zero for* $V_1^{-1}V_2$ *and* $V_2^{-1}V_1$ *to be MDS or NMDS. However, in Theorem 7.4, Theorem 7.5, and Theorem 7.6, each* x_i *needs to be nonzero; otherwise, the term* x_i^{-1} *in the conditions will not be defined.*

Remark 7.7. *We have presented a method for constructing involutory MDS and NMDS matrices using generalized Vandermonde matrices* $V_\perp(x; I)$ *with* $I = \{n - 1\}$. *However, we have not been able to determine the conditions for constructing involutory MDS and NMDS matrices from generalized Vandermonde matrices with* $I = \{1\}$ *and* $I = \{1, n\}$.

Remark 7.8. *This chapter does not consider the generalized Vandermonde matrices* $V(\mathbf{x}; T)$ *with discontinuities other than* $\{1\}$, $\{n - 1\}$, *or* $\{1, n\}$, *or those with more than two discontinuities. This is because the conditions for being MDS or NMDS matrices become more complicated. However, it is possible to find additional direct constructions of MDS and NMDS matrices by using Theorem 2.8.*

Till now, we have discussed nonrecursive constructions of MDS and NMDS matrices. In the next section, we will explore the recursive constructions of MDS and NMDS matrices using the direct method.

7.3 Direct Construction of Recursive MDS and NMDS Matrices

This section introduces several techniques for the direct construction of MDS and NMDS matrices over finite fields using recursive approach. To the best of our knowledge, we are the first to provide a direct construction method for recursive NMDS

matrices. We begin by establishing a criterion for determining the similarity between a companion matrix and a diagonal matrix. Using this condition, we can represent the companion matrix as a combination of a Vandermonde matrix and a diagonal matrix. We utilize determinant expressions for generalized Vandermonde matrices to present several techniques for constructing recursive NMDS matrices that are derived from companion matrices. Furthermore, a new direct construction for recursive MDS matrices is introduced.

Lemma 7.1. *Consider a monic polynomial $g(x) \in \mathbb{F}_q[x]$ of degree n with n distinct roots denoted as $\lambda_1, \ldots, \lambda_n \in \bar{\mathbb{F}}_q$. Then the matrix*

$$G' = \begin{bmatrix} 1 & \lambda_1 & \ldots & \lambda_1^{n-1} & \lambda_1^m & \lambda_1^{m+1} & \ldots & \lambda_1^{m+n-1} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \lambda_n & \ldots & \lambda_n^{n-1} & \lambda_n^m & \lambda_n^{m+1} & \ldots & \lambda_n^{m+n-1} \end{bmatrix} \tag{7.9}$$

is also a generator matrix for the $[2n, n]$ linear code $\mathcal{C}$ with generator matrix $G = [I \mid (C_g^T)^m]$.

Proof. From Theorem 2.10, we know that if a polynomial $g(x)$ has n distinct roots $\lambda_1, \ldots, \lambda_n$, then the companion matrix C_g associated to $g(x)$ can be written as $C_g = VDV^{-1}$, where

$$\begin{aligned} V &= vand(\lambda_1, \lambda_2, \ldots, \lambda_n) \\ &= \begin{bmatrix} 1 & 1 & \ldots & 1 \\ \lambda_1 & \lambda_2 & \ldots & \lambda_n \\ \lambda_1^2 & \lambda_2^2 & \ldots & \lambda_n^2 \\ \vdots & \vdots & \vdots & \vdots \\ \lambda_1^{n-1} & \lambda_2^{n-1} & \ldots & \lambda_n^{n-1} \end{bmatrix} \end{aligned}$$

and $D = diag(\lambda_1, \ldots, \lambda_n)$.

Consider an $[2n, n]$ linear code $\mathcal{C}$, with generator matrix $G = [I \mid (C_g^T)^m]$. Now

$$\begin{aligned} G &= [I \mid (C_g^T)^m] = [I \mid ((V^T)^{-1} D V^T)^m] \\ &= [I \mid (V^T)^{-1} D^m V^T] \\ &= (V^T)^{-1} [V^T \mid D^m V^T] \\ &= (V^T)^{-1} G', \end{aligned} \tag{7.10}$$

where $G' = [V^T \mid D^m V^T]$. Therefore, we have

$$G' = [V^T \mid D^m V^T]$$
$$= \begin{bmatrix} 1 & \lambda_1 & \dots & \lambda_1^{n-1} & \lambda_1^m & \lambda_1^{m+1} & \dots & \lambda_1^{m+n-1} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \lambda_n & \dots & \lambda_n^{n-1} & \lambda_n^m & \lambda_n^{m+1} & \dots & \lambda_n^{m+n-1} \end{bmatrix}.$$

Also, from (7.10), we have $G' = V^T G$. Hence, according to Lemma 2.5, we can conclude that G' is also a generator matrix for the linear code $\mathcal{C}$. □

Let C_g be the companion matrix associated with a monic polynomial $g(x)$ of degree $n \geq 3$. Then for $m < n$, it can be observed that the first row of C_g^m is a unit vector. Hence, the linear code generated by $[I \mid C_g^m]$ has minimum distance $< n$. Therefore, for $m < n$, C_g^m cannot be an MDS or NMDS matrix.

Theorem 7.7. *Consider a monic polynomial $g(x) \in \mathbb{F}_q[x]$ of degree n with n distinct roots, say $\lambda_1, \dots, \lambda_n \in \bar{\mathbb{F}}_q$. Let m be an integer satisfying $m \geq n$. Then, the matrix $M = C_g^m$ is MDS if and only if any set of n columns of the matrix G' defined in (7.9) is linearly independent.*

Proof. Based on Corollary 2.6, we can conclude that C_g^m is an MDS matrix if and only if its transpose $(C_g^m)^T = (C_g^T)^m$ is also an MDS matrix. Also, according to Definition 2.7, $(C_g^T)^m$ is MDS if and only if the $[2n, n]$ linear code $\mathcal{C}$, with generator matrix $G = [I \mid (C_g^T)^m]$, is an MDS code.

Now since $\lambda_1, \dots, \lambda_n$ are n distinct roots of $g(x)$, from Lemma 7.1, we can say that the matrix G' in (7.9) is also a generator matrix for the code $\mathcal{C}$. Therefore, by Remark 2.3, we can establish that $(C_g^m)^T$ is MDS, and hence C_g^m, if and only if any set of n columns of G' is linearly independent. □

Theorem 7.8. *Consider a monic polynomial $g(x) \in \mathbb{F}_q[x]$ of degree n with n distinct roots denoted as $\lambda_1, \dots, \lambda_n \in \bar{\mathbb{F}}_q$. Let m be an integer satisfying $m \geq n$. Then, the matrix $M = C_g^m$ is NMDS if and only if the matrix G' defined in (7.9) satisfies the three conditions specified in Lemma 2.4.*

Proof. From Corollary 2.6, we know that C_g^m is an NMDS matrix if and only if its transpose $(C_g^m)^T = (C_g^T)^m$ is also an NMDS matrix. Also, by Definition 2.11, $(C_g^T)^m$ is NMDS matrix if and only if the $[2n, n]$ linear code $\mathcal{C}$, with generator matrix $G = [I \mid (C_g^T)^m]$, is an NMDS code.

As $\lambda_1, \dots, \lambda_n$ are n distinct roots of $g(x)$, we can infer from Lemma 7.1 that the matrix G' defined in (7.9) is also a generator matrix for the code $\mathcal{C}$. Consequently,

we can conclude that $(C_g^m)^T$ is NMDS, and therefore C_g^m is NMDS, if and only if the matrix G' satisfy the three conditions outlined in Lemma 2.4. □

Lemma 7.2. *Suppose that* $g(x) = \prod_{i=1}^{n}(x - \lambda_i) \in \mathbb{F}_q[x]$ *yields a recursive MDS (NMDS) matrix. Then, for any* $c \in \mathbb{F}_q^*$, *the polynomial* $c^n g\left(\frac{x}{c}\right) = \prod_{i=1}^{n}(x - c\lambda_i)$ *also yields a recursive MDS (NMDS) matrix.*

Proof. Let $g^*(x) = c^n g\left(\frac{x}{c}\right)$. The matrix $C_{g^*} = cDC_gD^{-1}$ where

$$D = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 & 0 \\ 0 & c & 0 & \dots & 0 & 0 \\ 0 & 0 & c^2 & \dots & 0 & 0 \\ & & & \dots & & \\ 0 & 0 & 0 & \dots & c^{n-2} & 0 \\ 0 & 0 & 0 & \dots & 0 & c^{n-1} \end{bmatrix}$$

The matrix $C_{g^*}^m = c^m DC_g^m D^{-1}$ is MDS (NMDS) if and only if C_g^m is MDS (NMDS). □

Using the above lemma, it is possible to obtain more polynomials that produce recursive MDS or NMDS matrices from an initial polynomial.

Now, we introduce two approaches for constructing polynomials that yield recursive NMDS matrices. These polynomials are designed to have distinct roots. The underlying idea behind these techniques is based on Theorem 7.8: we carefully select suitable values for λ_i, for $1 \leq i \leq n$, and validate that the polynomial $g(x) = \prod_{i=1}^{n}(x - \lambda_i) \in \mathbb{F}_q[x]$ satisfies the conditions outlined in Theorem 7.8. To do so, we must examine the rank of the submatrices of G' constructed from any t columns (here we examine $t = n-1, n, n+1$) of G' corresponding to λ_i's as given in (7.9). A submatrix $G'[R]$, constructed from any t columns of G', is given by

$$G'[R] = \begin{bmatrix} \lambda_1^{r_1} & \lambda_1^{r_2} & \dots & \lambda_1^{r_t} \\ \lambda_2^{r_1} & \lambda_2^{r_2} & \dots & \lambda_2^{r_t} \\ \vdots & \vdots & \ddots & \vdots \\ \lambda_n^{r_1} & \lambda_n^{r_2} & \dots & \lambda_n^{r_t} \end{bmatrix}, \tag{7.11}$$

where R denotes a set $\{r_1, r_2, \dots, r_t\} \subset E = \{0, 1, \dots, n-1, m, m+1, \dots, m+n-1\}$ of t elements.

Theorem 7.9. *Consider the polynomial* $g(x) = \prod_{i=1}^{k}(x-\lambda_i)$, *where* $\lambda_i = \theta^{i-1}$ *for* $1 \leq i \leq n-1$ *and* $\lambda_n = \theta^n$ *for some* $\theta \in \mathbb{F}_q^*$. *Let* $E = \{0, 1, \ldots, n-1, m, m+1, \ldots, m+n-1\}$ *for some integer* $m \geq n$. *The matrix* C_g^m *is NMDS if and only if* $\theta^r \neq \theta^{r'}$ *for* $r, r' \in E$ *and* $\sum_{i=1}^{n} \theta^{r_i} = 0$ *for some* $R = \{r_1, r_2, \ldots, r_n\} \subset E$.

Proof. We have $\lambda_i = \theta^{i-1}$ for $1 \leq i \leq n-1$ and $\lambda_n = \theta^n$. So for $R = \{r_1, r_2, \ldots, r_t\} \subset E$ we have

$$G'[R] = \begin{bmatrix} 1 & 1 & \ldots & 1 \\ \theta^{r_1} & \theta^{r_2} & \ldots & \theta^{r_t} \\ \vdots & \vdots & \ddots & \vdots \\ (\theta^{n-2})^{r_1} & (\theta^{n-2})^{r_2} & \ldots & (\theta^{n-2})^{r_t} \\ (\theta^{n})^{r_1} & (\theta^{n})^{r_2} & \ldots & (\theta^{n})^{r_t} \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 1 & \ldots & 1 \\ \theta^{r_1} & \theta^{r_2} & \ldots & \theta^{r_t} \\ \vdots & \vdots & \ddots & \vdots \\ (\theta^{r_1})^{n-2} & (\theta^{r_2})^{n-2} & \ldots & (\theta^{r_t})^{n-2} \\ (\theta^{r_1})^{n} & (\theta^{r_2})^{n} & \ldots & (\theta^{r_t})^{n} \end{bmatrix}.$$

Now to prove the theorem, we can assume $x_{r_i} = \theta^{r_i}$ for $1 \leq i \leq t$ and apply Theorem 7.2. □

Example 7.11. *Consider the field* $\mathbb{F}_{2^4}$ *with the constructing polynomial* $x^4 + x + 1$ *and let* α *be a root of it. Let* $\theta = \alpha$. *We can verify that* $\theta^0 + \theta^1 + \theta^3 + \theta^7 = 0$. *Now, let us consider the polynomial* $g(x) = (x-1)(x-\alpha)(x-\alpha^2)(x-\alpha^4)$. *It can be verified that* C_g^m *is an NMDS matrix for* $4 \leq m \leq 11$.

Remark 7.9. *The above theorem assumes that* $\sum_{i=1}^{n} \theta^{r_i} = 0$ *for some* $R = \{r_1, r_2, \ldots, r_n\} \subset E$. *However, to ensure MDS property, the condition needs to be changed to* $\sum_{i=1}^{n} \theta^{r_i} \neq 0$ *for all* $R = \{r_1, r_2, \ldots, r_n\} \subset E$ *(refer to Theorem 3.18).*

Remark 7.10. *It can be observed that the condition imposed on* θ *in Theorem 7.9 remains valid when the values of* λ_i *are selected as* $\lambda_i = \theta^{i-1}c, 1 \leq i \leq n-1$, *and* $\lambda_n = \theta^n c$ *for a nonzero element* c *belonging to the field* $\mathbb{F}_q$. *By adopting this approach, the resulting polynomials are equivalent to those derived from the Lemma 7.2.*

Lemma 7.3. *Consider the polynomial* $g(x) = \prod_{i=1}^{k}(x-\lambda_i)$, *where* $\lambda_1 = 1$, *and* $\lambda_i = \theta^i$, $2 \leq i \leq n$, *for some* $\theta \in \mathbb{F}_q^*$. *Let* $E = \{0, 1, \ldots, n-1, m, m+1, \ldots, m+n-1\}$ *for some integer* $m \geq n$. *The matrix* C_g^m *is NMDS if and only if* $\theta^r \neq \theta^{r'}$ *for* $r, r' \in E$ *and* $\sum_{i=1}^{n} \theta^{-r_i} = 0$ *for some* $R = \{r_1, r_2, \ldots, r_n\} \subset E$.

Proof. Consider $\gamma_i = \lambda_{n-i+1} = (\theta^{-1})^{i-1}c, 1 \leq i \leq n-1$ and $\gamma_n = \lambda_1 = (\theta^{-1})^n c$ for $c = \theta^n$. Then by Theorem 7.9 and the above remark, the matrix C_g^m is NMDS if and only if $\theta^{-r_i}, 1 \leq i \leq n$, are distinct and $\sum_{i=1}^{n} \theta^{-r_i} = 0$ for some $R = \{r_1, r_2, \ldots, r_n\} \subset E$. Hence, the proof. □

Example 7.12. *Consider the field $\mathbb{F}_{2^4}$ with the constructing polynomial $x^4 + x + 1$ and let α be a root of it. Let $\theta = \alpha$. We can verify that $\theta^0 + \theta^{-1} + \theta^{-2} + \theta^{-7} = 0$. Now, let us consider the polynomial $g(x) = (x-1)(x-\alpha^2)(x-\alpha^3)(x-\alpha^4)$. It can be verified that C_g^m is an NMDS matrix for $4 \leq m \leq 11$.*

Remark 7.11. *The proof of the above lemma can also be seen similarly as in the proof of Theorem 7.9 by using Theorem 7.4.*

Remark 7.12. *The above lemma assumes that $\sum_{i=1}^{n} \theta^{-r_i} = 0$ for some $R = \{r_1, r_2, \ldots, r_n\} \subset E$. However, to ensure MDS property, the condition needs to be changed to $\sum_{i=1}^{n} \theta^{-r_i} \neq 0$ for all $R = \{r_1, r_2, \ldots, r_n\} \subset E$ (See Corollary 3.9).*

Now, we will present a direct construction of polynomial that yield recursive MDS matrix.

Theorem 7.10. *Consider the polynomial $g(x) = \prod_{i=1}^{k}(x - \lambda_i)$, where $\lambda_1 = 1$, $\lambda_i = \theta^i$, for $2 \leq i \leq n-1$, and $\lambda_n = \theta^{n+1}$ for some $\theta \in \mathbb{F}_q^*$. Let $E = \{0, 1, \ldots, n-1, m, m+1, \ldots, m+n-1\}$ for some integer $m \geq n$. The matrix C_g^m is NMDS if and only if $\theta^r \neq \theta^{r'}$ for $r, r' \in E$ and $(\sum_{i=1}^{n} \theta^{r_i})(\sum_{i=1}^{n} \theta^{-r_i}) - 1 \neq 0$ for all $R = \{r_1, r_2, \ldots, r_n\} \subset E$.*

Proof. We have $\lambda_1 = 1$, and $\lambda_i = \theta^i$ for $2 \leq i \leq n-1$ and $\lambda_n = \theta^{n+1}$. From Theorem 7.7, we know that the matrix C_g^m is MDS if and only if any n columns of G' are linearly independent. So for any $R = \{r_1, r_2, \ldots, r_n\} \subset E$ we have

$$G'[R] = \begin{bmatrix} 1 & 1 & \ldots & 1 \\ (\theta^2)^{r_1} & (\theta^2)^{r_2} & \ldots & (\theta^2)^{r_n} \\ \vdots & \vdots & \ddots & \vdots \\ (\theta^{n-1})^{r_1} & (\theta^{n-1})^{r_2} & \ldots & (\theta^{n-1})^{r_n} \\ (\theta^{n+1})^{r_1} & (\theta^{n+1})^{r_2} & \ldots & (\theta^{n+1})^{r_n} \end{bmatrix} = \begin{bmatrix} 1 & 1 & \ldots & 1 \\ (\theta^{r_1})^2 & (\theta^{r_2})^2 & \ldots & (\theta^{r_n})^2 \\ \vdots & \vdots & \ddots & \vdots \\ (\theta^{r_1})^{n-1} & (\theta^{r_2})^{n-2} & \ldots & (\theta^{r_{n-1}})^{n-2} \\ (\theta^{r_1})^{n+1} & (\theta^{r_2})^{n+1} & \ldots & (\theta^{r_n})^{n+1} \end{bmatrix}.$$

Let $y_{r_i} = \theta^{r_i}$ for $1 \leq i \leq n$. Therefore, we have

$$G'[R] = \begin{bmatrix} 1 & 1 & \dots & 1 \\ y_{r_1}^2 & y_{r_2}^2 & \dots & y_{r_n}^2 \\ \vdots & \vdots & \ddots & \vdots \\ y_{r_1}^{n-1} & y_{r_2}^{n-1} & \dots & y_{r_n}^{n-1} \\ y_{r_1}^{n+1} & y_{r_2}^{n+1} & \dots & y_{r_n}^{n+1} \end{bmatrix},$$

which is a generalized Vandermonde matrix of the form $V_\perp(\mathbf{y}; I)$ with $I = \{1, n\}$. Therefore, from Corollary 2.12 $\det(G'[R]) \neq 0$ if and only if y_{r_i} are distinct and $(\sum_{i=1}^n y_{r_i})(\sum_{i=1}^n y_{r_i}^{-1}) - 1 \neq 0$. Hence, the proof. □

Example 7.13. *Consider the field $\mathbb{F}_{2^4}$ with the constructing polynomial $x^4 + x + 1$ and let α be a root of it. Let $\theta = \alpha$ and consider the polynomial $g(x) = (x-1)(x-\alpha^2)(x-\alpha^3)(x-\alpha^5)$. It can be checked that the polynomial $g(x)$ satisfies the condition in Theorem 7.10, so it yields a recursive MDS matrix of order 4. It can be verified that C_g^4 is an MDS matrix.*

7.4 Conclusion

There has been significant research in the literature on the direct construction of MDS matrices using both recursive and nonrecursive methods. However, research on NMDS matrices has been limited in the literature, and there is currently no direct construction method available for them in a recursive approach. This chapter addresses this gap by presenting novel direct construction techniques for NMDS matrices in the recursive setting. By employing generalized Vandermonde matrices, we provide a new approach for constructing MDS and NMDS matrices. We also propose a method for constructing involutory MDS and NMDS matrices using generalized Vandermonde matrices. These direct constructions offer an efficient way of designing MDS and NMDS matrices, particularly for larger orders. Moreover, the chapter provides proof for some commonly referenced results related to the NMDS code. Overall, this work provides valuable tools for constructing MDS and NMDS matrices and advances the current state of research in this area.

8 FUTURE: A Lightweight Block Cipher with an Optimal Diffusion Matrix

Contents

8.1 Introduction

AES [DR02], SHA-256 [PUB15] and RSA [RSA78] are some of the most widely used cryptographic primitives, and they work well on systems with reasonable processing power and memory capabilities. But these primitives are not suitable in constrained environments such as RFID tags, sensor networks, contactless smart cards, medical services gadgets, etc. For this purpose in the recent decade, a large number of lightweight cryptographic primitives have been suggested and deployed on resource-constrained devices. Although there is no exact definition of lightweight cryptography, it is generally understood as a form of cryptography that places a strong emphasis on efficiency. Efficiency can be evaluated using various criteria such as hardware cost, power consumption, latency.

A block cipher converts plaintext blocks of a fixed length n (for the most part n= 64 or 128) to ciphertext blocks with the length n under the influence of a secret key k.

More specifically, a block cipher E is a function from $\mathbb{F}_2^n \times \mathbb{F}_2^k$ to $\mathbb{F}_2^n$ with the property that, for each key $K \in \mathbb{F}_2^k$, $E_K = E(\cdot, K)$ acts as a permutation on the elements of $\mathbb{F}_2^n$. Broadly, block ciphers can be arranged into two sorts: Feistel structure and substitution-permutation network (SPN) structure.

Both SPN and Feistel structures come with their own sets of advantages and disadvantages. Feistel structures (e.g. TWINE [SMMK13], Piccolo [SIH+11]) generally apply a round function to just one half of the block due to which they may be implemented in hardware with minimal cost. Additionally, implementing the inverse of Feistel constructions is not very challenging. This means that a circuit that handles both encryption and decryption functions can be designed with minimal extra complexity. However, as Feistel structures inject nonlinearity in just one half of the block in every round, such designs require more executions of round functions than SPN structures in order to preserve the *security margins* [1]. It is essential to note that these comparisons between SPN and Feistel entirely depends on the diffusion matrices (linear layer) and Sboxes utilized in the round function. For instance, DES [PUB77], which follows a Feistel structure, requires fewer rounds (DES has 16 rounds, whereas PRESENT has 31 rounds) than the SPN-based block cipher PRESENT [BKL+07]. However, when considering the same Sboxes and diffusion matrices in the round function, a Feistel structure requires more rounds than an SPN structure to achieve the desired security margin.

Due to the large deployment of low-resource devices and expanding need to provide security among such devices, lightweight cryptography has become a popular topic. Thus, research on designing and analyzing lightweight block ciphers has got a great deal of attention. Initial lightweight block ciphers such as NOEKEON [DPAR00], PRESENT [BKL+07] and KATAN [DCDK09] primarily emphasized minimizing the chip area and utilized simple round functions as their primary components.

With the advent of lightweight block ciphers, this field expanded dramatically in terms of possibilities. At this point, we have specialized ciphers that are optimized for code size, latency, energy and power. For example we have SIMON and SPECK [BSS+15] for code size, PRINCE [BCG+12] and MANTIS [BJK+16] for latency and MIDORI [BBI+15] and GIFT [BPP+17] for energy. Notably, NOEKEON [DPAR00] was designed with a hardware-oriented focus even before the term lightweight cryptography was coined.

[1] In terms of security, we often talk about a cipher's *security margin*. If a cipher has n rounds, and there is a cryptanalytic attack against the round-reduced version with $n-k$ rounds, the cipher has an absolute security margin of k rounds or a relative security margin of k/n [DR02].

Moreover, some block ciphers have optimized the cost of implementing decryption alongside encryption. For example, in MIDORI and NOEKEON, all components are involutory, and PRINCE exhibits the α-reflection property. Several lightweight block ciphers, including LED [GPPR11], MIDORI, and SKINNY [BJK+16], incorporate the overall structure of the AES round function and modify its elements to enhance their performance.

There have also been attempts to develop lightweight *tweakable block ciphers*, a block cipher equipped with an additional input known as a *tweak*. This feature enables enhanced encryption modes and facilitates the construction of efficient authenticated encryption. SKINNY, MANTIS, CRAFT [BLMR19], QARMA [Ava17] are some examples of such primitives. Also for CRAFT, design considerations were made to ensure that its implementations were resistant to Differential Fault Analysis (DFA) attacks.

In cryptography, *confidentiality* and *integrity* are two fundamental security goals. Confidentiality ensures that information is kept private and protected from unauthorized access. Encryption schemes provide this functionality: given a secret key, they convert plaintext into ciphertext. Decrypting the ciphertext to recover the original message should be infeasible unless the secret key is known. Whereas, integrity ensures that data remains unaltered during storage, transmission, or processing. To address the integrity problem, message authenticated codes (MACs) can be used. With a secret key, a MAC function produces a tag. Making a tag without the key (called forgery) should be practically impossible. MACs, similar to encryption methods, can be built using permutations, block ciphers, or tweakable block ciphers.

Confidentiality and integrity are often combined into a single security notion: authenticated encryption. This can be done by using a MAC to the plaintext and then encrypt them together (MAC-then-encrypt), encrypt the plaintext to get a ciphertext and append a MAC of the ciphertext (Encrypt-then-MAC) or by encrypt the plaintext and append a MAC of the plaintext (Encrypt-and-MAC). However, among these three composition methods, only Encrypt-then-MAC is considered as secure. There are also specific methods for this. In August 2018, the United States National Institute of Standards and Technology (NIST) issued a call for submissions to a standardization project focused on lightweight authenticated encryption.

In February 2019, NIST received 57 submissions for consideration. Out of these, 56 were accepted as first-round candidates in April 2019. After four months, NIST chose 32 candidates for the second round. In March 2021, NIST revealed 10 finalists, including ASCON [DEMS21], Elephant [BCDM21], GIFT-COFB [BCI+21],

Grain-128AEAD [HJM+21], ISAP [DEM+21], PHOTON-Beetle [BCD+21], Romulus [GIK+21], SPARKLE [BBS+21], TinyJAMBU [WH21], and Xoodyak [DHM+21], to advance to the final round of the selection process. On February 7, 2023, NIST declared the decision to standardize the ASCON family for lightweight cryptography applications. For a comprehensive overview of the evaluation criteria and selection process we refer to [TMC+23].

Permutation Based Cryptography: Cryptographic permutations, unlike block ciphers, are keyless public permutations designed to behave like random permutations. In recent years, they have gaining popularity alongside block ciphers. The keyless nature of cryptographic permutations eliminates the need for separate processing of the key and data input, making them more efficient in certain situations compared to block ciphers. This efficiency became particularly evident during the SHA-3 competition, where many proposed schemes were built on cryptographic permutations. The selection of the permutation-based Keccak sponge function [BDPA11] as the SHA-3 standard further reinforced the community's confidence in the advantages of this approach.

In 2007, Bertoni and colleagues introduced a cryptographic permutation-based sponge function [BDPA07], originally intended for hashing. Soon after, various effective methods for encryption, authentication, and authenticated encryption were created [MRV15, BDH+17, BDPVA10]. Today, constructions based on permutations have become a successful and fully established alternative to methods relying on block ciphers. Notably, Ascon [DEMS21], the winner in the NIST lightweight competition [TMC+23], is also permutation-based.

Tweakable Block Cipher: *Tweakable block ciphers* extend the concept of block ciphers by introducing an extra public input known as the *tweak*. The concept was formally introduced by Liskov et al. [LRW02, LRW11].

Definition 8.1. *A n-bit tweakable block cipher with k-bit key and t-bit tweak is a function*

$$\tilde{E}\colon \mathbb{F}_2^n \times \mathbb{F}_2^k \times \mathbb{F}_2^t \to \mathbb{F}_2^n,$$
$$(P, K, T) \mapsto C,$$

which maps an n-bit plaintext P to the n-bit ciphertext C using the secret key $K \in \mathbb{F}_2^k$ and tweak $T \in \mathbb{F}_2^t$.

The definition of a tweakable block cipher with parameters (n, k, t) and a classical

block cipher with parameters $(n, k+t)$ seems to have no significant difference. This is because a key-tweak pair (K, T) can be regarded simply as one element $K||T \in \mathbb{F}_2^{k+t}$, serving as the key. The main reason for distinguishing between tweakable block ciphers and classical block ciphers lies in keeping the key secret while assuming the tweak is public information, serving as a parameter to introduce variability to the actual instance.

The idea behind allowing this additional variability, as explained in [LRW11], is the need for variability at the mode of operation level. For example, in the CTR mode [Dwo01], a counter is used to vary the encryption functions in each block. The suggestion from the authors of [LRW11] is to directly incorporate the source of variability into the block cipher itself. As an illustration of a mode of operation, each block could be encrypted with the same tweakable block cipher, and different counters are incorporated as the tweaks. Such a tweakable block cipher should be designed to allow more efficient changes to the tweak than to the key.

Importance of Block Length in a Block Cipher: Modern ciphers commonly employ block size of $n = 64, 128$, or 256 bits. However, it should be noted that the block size n also serves as a crucial security parameter, determining the amount of data that can be securely encrypted using the same key. We typically expect block ciphers to be secure with up to 2^n queries. However, in many modes of operation (such as CBC, CFB, OFB [Dwo01], etc.), security diminishes significantly beyond $\sigma = 2^{n/2}$ blocks of message, a limit known as the birthday bound. Consequently, while birthday bound attacks are of minimal concern when using block ciphers with a block size of $n = 128$ bits, they pose a serious concern when employing a block cipher with $n = 64$ bits, requiring relatively frequent rekeying to keep $\sigma \ll 2^{32}$ [Rog11, Section 4.5]. This attack scenario is not purely theoretical, as highlighted by the authors in [BL16].

A Challenge in Lightweight Block Cipher: MDS matrices are widely recognized for their ability to provide maximum diffusion in block ciphers. However, in the context of lightweight block ciphers, MDS matrices are often avoided in the round function due to their high implementation cost. As a result, lightweight block ciphers typically require more rounds to achieve a desired level of security against well-known attacks such as differential [BS91a, BS91b], impossible differential [BBS99, BBS05], and linear attacks [Mat94]. Therefore, incorporating MDS matrices into a lightweight block cipher poses a significant challenge.

To address this challenge, this chapter introduces a new block cipher called FUTURE. FUTURE overcomes this challenge by judiciously choosing a very lightweight MDS matrix, which is constructed as a composition of four sparse matrices. Furthermore, FUTURE utilizes a lightweight yet cryptographically significant Sbox, which is formed by combining four different Sboxes.

Outline: The rest of this chapter is structured as follows: Section 8.2 provides a detailed discussion on the specification of FUTURE. Section 8.3 examines the design decisions made in selecting the components of FUTURE. Section 8.4 presents a security analysis of FUTURE. Section 8.5 delves into the implementation cost of FUTURE. Lastly, Section 8.6 concludes the chapter.

8.2 Structure of FUTURE

FUTURE is a new SPN-based block cipher and consists of 10 rounds. It accepts 128-bit keys and has a block size of 64-bit. FUTURE has been designed with a specific focus on minimizing latency and reducing hardware implementation costs when implemented in fully unrolled setting, where the entire encryption is performed in a single clock cycle.

8.2.1 Round Function

The encryption round of FUTURE consists of four distinct transformations applied in the following sequence: SubCell (SubByte), MixColumn, ShiftRow, and AddRoundKey (as depicted in Figure 8-1). It is worth mentioning that the terms SubCell, MixColumn, ShiftRow, and AddRoundKey were first introduced in the block cipher AES [DR02], and nowadays, they have become so standard for describing their functionalities. We also adopt these terms from AES for the round function of FUTURE. The final round of FUTURE is slightly different from the other nine rounds, MixColumn operation is removed here. Since FUTURE is an AES-like cipher and, as mentioned in [DR02, Section 10.2.3], the addition of a MixColumn operation in the last round does not enhance the security, we have opted to remove the MixColumn operation in the last round.

The cipher receives a 64-bit plaintext $P = b_0 b_1 b_2 \dots b_{62} b_{63}$ as the cipher state I, where b_0 is the most significant bit [2]. The cipher state can be represented as 16 4-bit

[2]In a binary number, each bit holds a value that is a power of 2. The most significant bit (MSB) represents the highest power of 2 in the number. For example, in a 4-bit binary number $x_0x_1x_2x_3$ where x_3 is the MSB, the corresponding decimal value is calculated as $x_0 \cdot 2^3 + x_1 \cdot 2^2 + x_2 \cdot 2^1 + x_3 \cdot 2^0$.

Table 8.1: Specifications of FUTURE Sbox.

x		0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
S_4(x)		0	1	2	3	4	5	6	7	8	9	e	f	c	d	a	b
S_3(x)		0	1	2	3	4	d	6	f	8	9	a	b	c	5	e	7
S_2(x)		1	3	0	2	5	7	4	6	9	a	8	b	d	e	c	f
S_1(x)		0	1	2	3	4	7	6	5	8	9	a	b	c	f	e	d
S(x)		1	3	0	2	7	e	4	d	9	a	c	6	f	5	8	b

cells as follows:

$$I = \begin{bmatrix} s_0 & s_4 & s_8 & s_{12} \\ s_1 & s_5 & s_9 & s_{13} \\ s_2 & s_6 & s_{10} & s_{14} \\ s_3 & s_7 & s_{11} & s_{15} \end{bmatrix},$$

i.e. $s_i = b_{4i}b_{4i+1}b_{4i+2}b_{4i+3} \in \{0,1\}^4$ for $0 \leq i \leq 15$. The input state for the i-th round, denoted as I_i, is defined such that $I_0 = P$.

Nonlinear Transformation SubCell: SubCell is a nonlinear transformation that applies a 4-bit Sbox S to each cell of the internal state of the cipher.

$$s_i \leftarrow S(s_i) \quad \text{for } i = 0, 1, \ldots, 15.$$

The Sbox S is a composition of four low hardware cost Sboxes S_1, S_2, S_3 and S_4 i.e. $S(s_j) = S_1 \circ S_2 \circ S_3 \circ S_4(s_j)$ for $j = 0, 1, \ldots, 15$.

The Sboxes in hexadecimal notation are given by the following Table 8.1.

Linear Transformation MixColumn: The MixColumn is a linear operation that operates separately on each of the four columns of the state. FUTURE uses an MDS matrix M for the MixColumns operation. We have

$$(s_i, s_{i+1}, s_{i+2}, s_{i+3})^T \leftarrow M \cdot (s_i, s_{i+1}, s_{i+2}, s_{i+3})^T$$

for $i = 0, 4, 8, 12$.

Where M is an MDS matrix given by

$$M = \begin{bmatrix} \alpha^3 & \alpha^3+1 & 1 & \alpha^3 \\ \alpha+1 & \alpha & \alpha^3+1 & \alpha^3+1 \\ \alpha & \alpha+1 & \alpha^3 & \alpha^3+1 \\ \alpha^3+1 & \alpha^3+1 & \alpha^3 & 1 \end{bmatrix}$$

which is constructed by composition of 4 sparse matrices M_1, M_2, M_3 and M_4 of order 4 i.e. $M = M_1M_2M_3M_4$, where

$$M_1 = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, M_2 = \begin{bmatrix} 0 & 0 & 1 & \alpha \\ 1 & 0 & 0 & 0 \\ \alpha^3+1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, M_3 = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ \alpha^3+1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \text{ and } M_4 = M_1 \tag{8.1}$$

The multiplications between matrices and vectors are performed over $\mathbb{F}_{2^4}$ defined by the primitive polynomial $x^4 + x + 1$ and α is a root of this polynomial.

Cell Permutation ShiftRow: ShiftRow rotates row i of the array state i cell positions to the right for $i = 0, 1, 2, 3$. We have,

$$\begin{bmatrix} s_0 & s_4 & s_8 & s_{12} \\ s_1 & s_5 & s_9 & s_{13} \\ s_2 & s_6 & s_{10} & s_{14} \\ s_3 & s_7 & s_{11} & s_{15} \end{bmatrix} \leftarrow \begin{bmatrix} s_0 & s_4 & s_8 & s_{12} \\ s_{13} & s_1 & s_5 & s_9 \\ s_{10} & s_{14} & s_2 & s_6 \\ s_7 & s_{11} & s_{15} & s_3 \end{bmatrix}.$$

i.e. $s_i \leftarrow 13 \cdot s_i \pmod{16}$ for $i = 0, 1, \ldots, 15$.

Note that in the ShiftRow operation of AES [DR02] and LED [GPPR11], the row i of the array state is rotated i cell positions to the left, for $i = 0, 1, 2, 3$.

AddRoundKey: The i-th round key RK_i for $1 \leq i \leq 10$ is XORed with the state I.

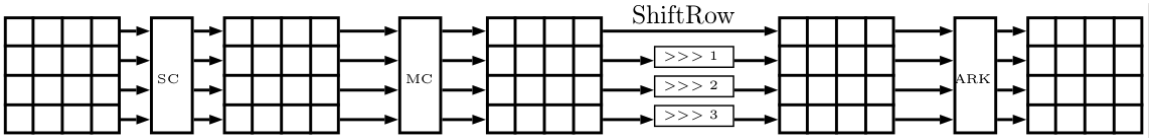
Data Processing: The encryption process of FUTURE involves data processing through a total of 10 rounds. The encryption function F takes a 64-bit data $X \in \{0,1\}^{64}$, whitening keys $WK \in \{0,1\}^{64}$ and 10 round keys $RK_i \in \{0,1\}^{64}$ $(1 \leq i \leq 10)$ as the inputs and outputs a 64-bit data $Y \in \{0,1\}^{64}$. F is defined as follows:

$$F = \begin{cases} \{0,1\}^{64} \times \{0,1\}^{64} \times \left\{\{0,1\}^{64}\right\}^{10} \rightarrow \{0,1\}^{64} \\ (X, WK, RK_1, RK_2, \ldots, RK_{10}) \rightarrow Y. \end{cases}$$

ALGORITHM 1: Encryption Function of FUTURE.

Input: X and $WK, RK_1, RK_2, \ldots, RK_{10}$
Initialization: $S \leftarrow$ **KeyAdd**(X, WK) ;
for $i \leftarrow 1$ **to** 9 **do**
  $S \leftarrow$ **SubCell**(S);
  $S \leftarrow$ **MixColumn**(S);
  $S \leftarrow$ **ShiftRows**(S);
  $S \leftarrow$ **AddRoundKey**(S, RK_i);
end
$S \leftarrow$ **SubCell**(S);
$S \leftarrow$ **ShiftRows**(S);
$Y \leftarrow$ **AddRoundKey**$(S, RK10)$;
Output: Y

Figure 8-1: The round function applies four different transformations: SubCell (SC), MixColumn (MC), ShiftRow and AddRoundKey (ARK).



The Round Key Evolution and round constants: FUTURE uses a 128-bit secret key $K = k_0 k_1 \ldots k_{127}$. It splits K in two equal parts K_0 and K_1 for the round key and whitening key generation i.e. $K = K_0 || K_1$, where $K_0 = k_0 k_1 \ldots k_{63}$ and $K_1 = k_{64} k_{65} \ldots k_{127}$ are two 64-bit keys. It uses K_0 as whitening key and the round key RK_i $(1 \leq i \leq 10)$ generation is as follows (see Figure 8-2):

$$RK_i = \begin{cases} K_0 \leftarrow K_0 \lll (5 \cdot \frac{i}{2}) & \text{if } 2 \mid i \\ K_1 \leftarrow K_1 \lll (5 \cdot \lfloor \frac{i}{2} \rfloor) & \text{if } 2 \nmid i \end{cases}$$

where $K_i \lll j$ means the 64-bit word obtained by a j-bit left rotation (left cyclic shift) of K_i.

For FUTURE a single bit "1" is XORed into each 4-bit cell (in different positions) of every round except the 5th and 10th round. The round constants are defined as shown in Table 8.2.

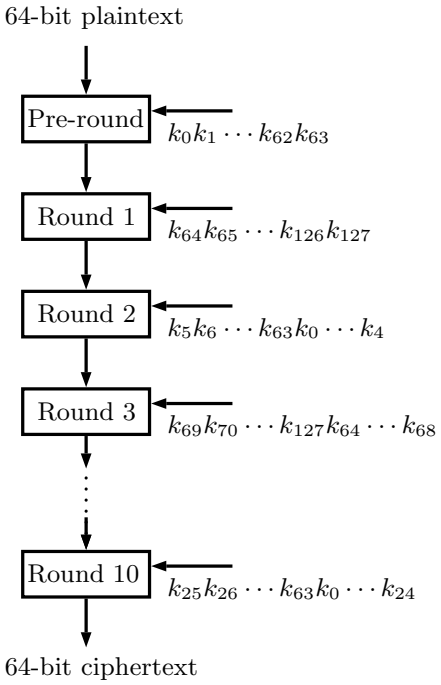
More specifically, we are adding a NOT gate in each cell except 5th and 10th round.

In the following section we justify the decisions we took during the design of FUTURE.

Table 8.2: The round constants (in hexadecimal) for the N-th round of FUTURE.

Rounds (N)	**Round constant**
1, 6	0x1248248148128124
2, 7	0x2481481281241248
3, 8	0x4812812412482481
4, 9	0x8124124824814812
5, 10	0x0000000000000000

Figure 8-2: Round Key Generation.



8.3 Design Decision

The design choice of round function for FUTURE has been inspired by the existing block ciphers, however all the components of FUTURE are new. Sometimes it is preferred to use an SPN-based block cipher over a Feistel-based one, as round function in Feistel-based block ciphers operates on just half of the state, which results in more rounds for encryption. Moreover, since the AES-like design has a simple structure and is specifically designed using the wide-trail strategy, we have chosen FUTURE to be an AES-like cipher (Definition 2.47). However, it is essential to note that the operations in the round function of FUTURE do not follow the sequence found in the round function of a AES-like cipher. Specifically, in FUTURE, MixColumn occurs

before ShiftRow in the round function.

Also, an unrolled implementation offers the best performance due to the computation of full encryption within one clock cycle. It does have the disadvantage of extending the critical path since the encryption or decryption operation is implemented as a combinatorial circuit in its entirety. However, in this implementation, there is no requirement for registers to hold the intermediate states. This means a low implementation cost with a small delay for block ciphers with a small number of rounds. Since FUTURE requires only 10 rounds for full encryption, we have deemed it more suitable for implementation in a fully unrolled circuit, though it can also be implemented in a rolled fashion.

8.3.1 SubCell

As the only nonlinear operation in the FUTURE, Sbox plays a significant role against various attacks. To increase the cipher's resistance to linear cryptanalysis [Mat94] and differential cryptanalysis [BS91b], any n-bit Sbox should have small magnitude entries in the linear approximation table (LAT) and difference distribution table (DDT) respectively, excluding the first entry in the first row. In other words, the maximal absolute bias of a linear approximation and the maximal probability of a differential of an Sbox should be low. Also, the cost of the Sbox, including its area and critical path, constitutes a substantial portion of the overall cost. Therefore, the selection of an Sbox that optimizes these expenses is crucial in the design of a lightweight block cipher.

For the SubCell operation, we use a 4-bit Sbox that is extremely efficient in terms of hardware and also meets the following criteria:

1. Nonlinearity of the Sbox is 4 (which is optimal).
2. The maximal probability of a differential is 2^{-2} and there are exactly 24 differentials with probability 2^{-2}.
3. The maximal absolute bias of a linear approximation is 2^{-2} and there are exactly 36 linear approximations with absolute bias 2^{-2}.
4. There is no fixed point.

FUTURE Sbox S is a composition of four Sboxes S_1, S_2, S_3 and S_4 (See Table 8.1). The algebraic normal form of the coordinate Boolean functions of S is given by

$$l_3(x) = x_0x_1x_3 \oplus x_0x_2 \oplus x_3$$
$$l_2(x) = x_1x_3 \oplus x_2$$
$$l_1(x) = x_0x_2x_3 \oplus x_0x_2 \oplus x_0 \oplus x_1x_2 \oplus x_2$$
$$l_0(x) = x_0x_1x_3 \oplus x_0x_2 \oplus x_0x_3 \oplus x_1 \oplus 1.$$

Thus, we can see that the maximal and minimal algebraic degree of S are 3 and 2 respectively.

To find lightweight 4-bit Sboxes, we chose to explore circuits systematically from the bottom-up approach, starting with the identity function's circuit (or by bit wiring of the circuit) and adding gates sequentially. We have decided to choose only NAND, XOR, and XNOR gates as some popular block ciphers like SKINNY [BJK+16] and Piccolo [SIH+11] use lightweight 4-bit Sboxes that can be implemented by a minimum number of these logic gates. First, we have searched for the circuits representing a 4-bit Sbox that can be implemented by (i) one XOR/XNOR gate or by (ii) one NAND gate followed by one XOR/XNOR gate. As a result, we have the two sets of 4-bit Sboxes, T_1 and T_2 [3], where T_1 contains the Sboxes implemented by one XOR/XNOR gate and T_2 contains the Sboxes implemented by one NAND and one XOR/XNOR gate. Next, we search for the Sboxes with low hardware cost and good cryptographic properties by composition of 2, 3 or 4 different Sboxes from the set $T_1 \cup T_2$. We obtain the FUTURE Sbox which is a composition of 4 Sboxes with 4 NAND, 3 XNOR and 1 XOR gates with is the lowest hardware cost for our search of 4-bit Sboxes with the optimal nonlinearity of 4.

During the search of an Sbox for FUTURE with this composition method, we only concentrate on the nonlinearity of the resulting Sbox. The nonlinearity of the Sboxes S_1, S_2, S_3 and S_4 are zero, whereas the resulting Sbox S has 4, which is the maximum value for a balanced 4-bit Sbox. The main concern for choosing such a composition method was to reduce implementation cost for the Sbox S. The hardware cost for S_1, S_2, S_3 and S_4 are very low. More specifically, they can be implemented with 4 NAND, 3 XNOR, and 1 XOR gates only (see Figure: 8-4,8-5,8-6 and 8-7), resulting in a low hardware cost (12 GE in UMC 180nm 1.8 V [UMC04]) for the Sbox S.

With this method, the implementation cost of an Sbox with the standard Sbox criteria (like balancedness, maximum nonlinearity, small value of δ_S and L_S etc.) may

[3]Some Sboxes of the sets T_1 and T_2 are given in Appendix A.4 and A.5. Also, note that we are not doing an exhaustive search to find all such Sboxes. More specifically, we have taken only 24 such elements from both T_1 and T_2, and the Sboxes S_1, S_2, S_3, and S_4, used to construct the FUTURE Sbox, are taken from the 48 Sboxes.

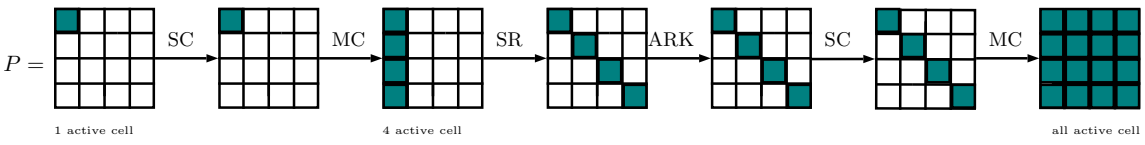
be reduced significantly. We believe that this is the first time to use this composition type Sbox with maximum nonlinearity and some useful cryptographic properties.

Also, it is worth mentioning that the 4-bit Sboxes used in SKINNY and Piccolo have the same nonlinearity and hardware cost as the FUTURE Sbox. But FUTURE Sbox is the new one and it is constructed by the composition of four lightweight Sboxes with zero nonlinearity. Also, it is not always trivial to get an Sbox with good cryptographic properties by the composition of four such lightweight Sboxes. We decided to use the newly constructed Sbox.

8.3.2 MixColumn

An NMDS matrix or a matrix with an even lower branch number (see Definitions 2.8 and 2.9) offers efficient implementation characteristics as compared to MDS matrices. However, due to their incorporation, the diffusion speed in the block cipher may be slower. The diffusion speed is determined by the number of rounds required to achieve *full diffusion*. Here, by full diffusion in a block cipher, we refer to the property that every state bit depends on all state bits two rounds ago, or a change in one state bit is likely to affect half of the state bits after two rounds [DR02, Section 3.5]. For a detailed overview of full diffusion in block cipher we refer to [Bei18, DR02]. Note that FUTURE requires only 2 rounds for the full diffusion (See Figure 8-3). Also, it is noteworthy that achieving full diffusion after 2 rounds is possible even when considering a matrix of order 4 with all nonzero entries. However, the selection of the MDS matrix in FUTURE is not solely based on the criterion of full diffusion. It also accounts for resistance against fundamental attacks, such as differential cryptanalysis [BS91a, BS91b] and linear cryptanalysis [Mat94]. This is because the minimum number of active Sboxes in a block cipher (designed based on the wide-trail strategy) with an MDS matrix in each round is higher compared to ciphers that utilize a matrix with a low branch number as part of MixColumn.

Figure 8-3: Full diffusion of FUTURE.



MDS matrices are not sparse. But they can be constructed from sparse matrices by recursive method. However, the MDS matrix in FUTURE is a composition of 4

different lightweight sparse matrices M_1, M_2, M_3 and M_4 (see 8.1 of Section 8.2.1). These matrices are of the form

$$\begin{bmatrix} 0 & 0 & m_1 & m_2 \\ m_3 & 0 & 0 & 0 \\ m_4 & m_5 & 0 & 0 \\ 0 & 0 & m_6 & 0 \end{bmatrix},$$

where $m_i \in \mathbb{F}_{2^4}$ for $i = 1, 2, \ldots, 6$.

It is worth mentioning that these matrices correspond to the GDLS matrices as defined in Definition 5.2. Specifically, these matrices can be expressed as $GDLS(\rho_1, \rho_2; D_1, D_2)$, where $\rho_1 = [2, 3, 4, 1]$, $\rho_2 = [3, 2, 1, 4]$, $D_1 = diag(m_3, m_5, m_6, m_2)$, and $D_2 = diag(m_4, 0, m_1, 0)$.

The idea of constructing MDS matrices in such a fashion was first introduced in [SM21] and we are the first to take advantage of this method in the design of FUTURE. In order to narrow down the search space for finding an MDS matrix using this approach, we set $m_1 = m_3 = m_6 = 1$ and conduct an exhaustive search ($5^{12} \approx 2^{28}$ choices) over the set $\{1, \alpha, \alpha^2, \alpha^{-1}, \alpha^{-2}\}$ to obtain $M = M_1 M_2 M_3 M_4$ as an MDS matrix.

The implementation cost for the MDS matrix M is minimized due to the low implementation cost of M_1, M_2, M_3 and M_4. Note that to construct MDS matrix in this method, the implementation cost is calculated by the sum of the implementation cost of M_1, M_2, M_3 and M_4.
We will now demonstrate how selecting specific elements from a finite field constructed by a specific irreducible polynomial improves multiplication efficiency.

The Primitive Polynomial $x^4 + x + 1$: The multiplications between the matrices M_1, M_2, M_3 and M_4 and vectors are performed over the field $\mathbb{F}_{2^4}$ constructed by the primitive polynomial $x^4 + x + 1$. The entries of these matrices are from the set $\{0, 1, \alpha, \alpha^3 + 1 = \alpha^{-1}\}$, where α is the primitive element of the field and $\alpha^4 + \alpha + 1 = 0$.

In $\mathbb{F}_{2^4}$, any element b can be expressed as $b = b_0 + b_1 \cdot \alpha + b_2 \cdot \alpha^2 + b_3 \cdot \alpha^3$. Then by the multiplication of b by $\alpha^3 + 1$ we have

$$(b_0 + b_1 \cdot \alpha + b_2 \cdot \alpha^2 + b_3 \cdot \alpha^3) \cdot (\alpha^3 + 1) = (b_0 + b_1) + b_2 \cdot \alpha + b_3 \cdot \alpha^2 + b_0 \cdot \alpha^3.$$

Thus, in vector form the above product looks like $(b_0 \oplus b_1,\ b_2,\ b_3,\ b_0)$, in which there is 1 XOR. Therefore, the XOR count of $\alpha^3 + 1$ is 1.

Similarly, we have $b \cdot \alpha = (b_3,\ b_0 \oplus b_3,\ b_1,\ b_2)$. Hence, the XOR count of α is 1. Also, the XOR count of 1 is 0 and there is no other nonzero element in the field with an XOR count of ≤ 1.

Thus, for the suitable choice of the constructing polynomial and entries of the matrices, the implementation cost of the MDS matrix M is reduced significantly. More specifically, FUTURE requires 35 XORs for the implementation of the MDS matrix (See Section 8.5.2).
The following table provides a comparison of the cost of the FUTURE MDS matrix with the matrices [4] used in the linear layer of some popular block ciphers.

Table 8.3: Comparison of cost of the Linear layers.

Block Cipher	**Linear Layer**	**Cost**
AES	MDS matrix	108 XORs
LED	Recursive MDS matrix	14 XORs
FUTURE	MDS matrix	35 XORs
Piccolo	MDS matrix	52 XORs
PRINCE	$(M^{(0)}, M^{(1)})$ NMDS matrix	24 XORs
MIDORI	NMDS matrix	24 XORs
SKINNY	Binary matrix with branch number 2	12 XORs
CRAFT	Binary matrix with branch number 2	12 XORs
PRESENT	Bit permutation	0
GIFT	Bit permutation	0

From Table 8.3, we can see that PRINCE and MIDORI use an NMDS matrix with a low implementational cost of 24 XORs. But for achieving security against various attacks they need more rounds than FUTURE. The linear layers in PRESENT and GIFT are a bit permutation of the state. As a result, the linear layer is created with simple wire shuffling and requires no hardware. But for resisting some fundamental attacks like linear cryptanalysis, differential cryptanalysis etc., they need a larger number of rounds than MIDORI and PRINCE. For the case of SKINNY and CRAFT, the binary matrix is of branch number 2 and needs only 12 XORs for implementation. For this, they attain full diffusion after 6th and 7th rounds respectively, which is 2 for FUTURE. While the cost of implementing the MDS matrix in LED is relatively low, it is worth noting that the companion matrix needs to be applied four times, resulting in a requirement of four clock cycles to obtain the MDS matrix. On the other hand, the FUTURE MDS matrix M is implemented in a single clock cycle with 35 XORs gates. Since FUTURE is designed to be more suitable for a fully unrolled implementation, M is preferred over the others in terms of XOR gates and security

[4]Each matrix in the Table 8.3 has an input size of 16 bits, except for the AES, which has an input size of 32 bits.

Table 8.4: The minimum number of active Sbox for N rounds of FUTURE.

Rounds (N)	1	2	3	4	5
Differential cryptanalysis	1	5	9	25	26
Linear cryptanalysis	1	5	9	25	26

parameters.

8.3.3 Round Key

For the key scheduling, we are mainly concerned about reducing hardware costs. Note that the key scheduling function in FUTURE is implemented as a bit permutation of the master key. It is, therefore, possible to create this module through simple wire shuffling and it takes up no hardware cost.

8.4 Security Analysis

The security of FUTURE against various cryptanalysis techniques is discussed in this section.

8.4.1 Differential and Linear Cryptanalysis

The most frequent and fundamental security analysis of a block cipher is to determine a cipher's resistance to differential and linear cryptanalysis (see Section 2.8.4). In this study, we utilized Mixed Integer Linear Programming (MILP) to determine lower limits for the minimum number of active Sboxes involved in differential and linear characteristics for different numbers of rounds. The obtained results [5], presented in Table 8.4. However, it is worth mentioning that in [DR02, Theorem 9.5.1], it is proven that in a four-round trail of AES, there will be at least 25 active Sboxes. This proof also holds for FUTURE. Nevertheless, in our analysis, we employed the bit-based MILP to precisely determine the differential and linear trail, enabling us to determine the actual differential probability and correlation potential from the DDT (Table A.1) and LAT (Table A.2) of FUTURE Sbox respectively.

Differential cryptanalysis: If $2^{-\delta}$ be the maximum probability of the differential propagation in a single Sbox and N_s be the number of active Sboxes in a

[5]Here we have done the bit-based MILP and we could not find any solution for the higher number of rounds ($n \geq 6$) by the MILP model due to its long-running time.

differential characteristic, then attack with the differential characteristic of a block cipher becomes infeasible if N_s satisfies the following condition [SSL15, Section 4.2.12]:

$$2^{\delta \cdot N_s} > 2^b \iff \delta \cdot N_s > b.$$

where b is the bit-length of the block size of the block cipher. For FUTURE, $b = 64$ and $\delta = 2$ and hence we must have $N_s > 32$. This is obtained by at most 7 rounds[6] for FUTURE.

However, for the 4-round FUTURE, we have searched 50 different single characteristics with the minimum number of active Sboxes (which is 25) with no Sbox activity pattern. Here we have observed that among these 50 characteristics the highest probability is 2^{-62}. Next we have fixed the input and output differences of the characteristic with highest probability and search for different single characteristics with the same Sbox activity pattern. Here we have found that only 2 characteristics are possible and the highest probability is also 2^{-62}. Also, from Table A.1, we can observe that there are only 24 differentials with probability 2^{-2} and whereas there are 72 differentials with probability 2^{-3}. Hence, we expect that the probability of any possible differential characteristic will be less than 2^{-63} when employing 5 rounds of FUTURE. Thus, we believe that full rounds of FUTURE are strong enough to resist differential cryptanalysis.

Linear cryptanalysis: Given a linear characteristic with a bias ϵ, $4\epsilon^2$ is defined as the correlation potential. For an adversary to perform linear cryptanalysis on an n-bit block cipher, the correlation potential must be more than 2^{-n}.

Similar to the differential, for the 4th round of FUTURE, we have searched 50 different single linear characteristics with the minimum number of active Sboxes with no Sbox activity pattern. Among which, the highest correlation potential is 2^{-74}. Next, with the same input and output masking of the highest correlation potential, we proceed to explore an additional set of 10 distinct single characteristics that exhibit the same Sbox activity pattern. We observe that it has a linear hull effect (average correlation potential) of $2^{-73.66}$. Also, from Table A.2, we can observe that there are exactly 36 linear approximations with absolute bias 2^{-2} and whereas there are 96 linear approximations with absolute bias 2^{-3}. So we expect that for 5-round FUTURE, the correlation potential will be lower than 2^{-64}. Hence, we believe that 10-round FUTURE provides adequate resistance against linear cryptanalysis.

[6]Since the minimum number of active Sboxes in round 3 and round 4 are 9 and 25 respectively, for 7th round FUTURE there will be at least 34 active Sboxes.

8.4.2 Impossible Differential Attacks

In classical differential cryptanalysis, the focus is on finding a differential with a notably high probability. The suggested key candidate is the one with the highest observed occurrence of the output difference. *Impossible differential cryptanalysis*, introduced by Knudsen [Knu98] and Biham et al. [BBS99, BBS05], works in the opposite way. It discards any potential keys that would result in an output difference that is already known to be impossible.

An *impossible differential* in an encryption function F is defined by a differential $(\Delta x, \Delta y)$, where for all plaintexts x, the equation $F(x) + F(x + \Delta x) \neq \Delta y$ holds. By exploiting such a difference in a reduced round version of the cipher, it is possible to launch a key recovery attack on the cipher in some more rounds. The attack involves selecting an adequate number of plaintexts with input differences that align with the impossible differential and gathering the corresponding ciphertexts. Subsequently, through the partial decryption of additional rounds using all potential subkeys, we can eliminate those subkeys that lead to impossible differentials.

FUTURE requires 2 rounds to achieve full diffusion after 2 rounds in both forward and backward. So we expect that there is no certain impossible differential characteristic over 4 rounds. To find the actual impossible differential characteristics, we employ the MILP method proposed in [CCJ+16, ST17] while taking into account the DDT of the Sbox in the FUTURE. Specifically, we thoroughly test input and output differences that meet the following criteria similar to [BPP+17]:

1. The input difference activates only one of the first 4 Sboxes.
2. The output difference activates only one of the 16 Sboxes.

In the first case, there are a total of $4 \times 15 = 60$ possible input differences meeting these conditions. In the second case, there are $16 \times 15 = 240$ possible output differences. Consequently, we examined a total of $14,400$ pairs of input and output differences.

The results of our search revealed that for a 4-round implementation of FUTURE, only 267 out of the $14,400$ pairs exhibited impossible differentials. We then extended this search to 5 rounds and found that there were no impossible differentials among the $14,400$ pairs. So we expect that full rounds of FUTURE are strong enough to resist the impossible differential attack.

8.4.3 Boomerang Attack

The boomerang attack [Wag99] is a type of differential attack in which the attacker does not attempt to cover the entire block cipher with a single differential characteristic with high probability. Instead, the boomerang attack strategy involves dividing the cipher into two sub-ciphers and focusing on finding a boomerang quartet with high probability. The probability of constructing a boomerang quartet is denoted as $\hat{p}^2\hat{q}^2$, where

$$\hat{p} = \sqrt{\sum_{\beta} \Pr^2[\alpha \to \beta]}$$

and α and β are input and output differences for the first sub-cipher and $\hat{q}$ for the second sub-cipher. This attack is effective when an n-bit cipher satisfies $\hat{p}^2\hat{q}^2 \leq 2^{-n}$.

The value of $\hat{p}^2$ is bounded by the maximum differential characteristic probability, i.e., $\hat{p}^2 \leq \max_{\beta} \Pr[\alpha \to \beta]$. The same bound applies to $\hat{q}^2$ as well. Let p and q denote the maximum differential trail probabilities for the first and second sub-ciphers, respectively. It is known that p and q are bounded by $2^{-2 \cdot N_s}$, where N_s is the minimum number of active Sboxes in each sub-cipher. By referring to Table 8.4, we observe that any combination of two sub-ciphers in an 8-round FUTURE has at least 32 active Sboxes in total. Hence, we conclude that the full round of FUTURE is secure against boomerang attacks.

8.4.4 Integral Attack

We first search for integral distinguishers for the round reduced versions of FUTURE by using the (bit-based) division property [TM16] and using the Mixed-Integer Linear Programming approach described in [SWW20, XZBL16]. We first evaluate the propagation of the division property for the Sbox. The algebraic normal form of FUTURE Sbox is given by

$$\begin{aligned}
y_3 &= x_0x_1x_3 \oplus x_0x_2 \oplus x_3 \\
y_2 &= x_1x_3 \oplus x_2 \\
y_1 &= x_0x_2x_3 \oplus x_0x_2 \oplus x_0 \oplus x_1x_2 \oplus x_2 \\
y_0 &= x_0x_1x_3 \oplus x_0x_2 \oplus x_0x_3 \oplus x_1 \oplus 1.
\end{aligned}$$

and the propagation of the division property is summarized as Table 8.5.

Here, let u and v be the input and output division property, respectively. The propagation from u to v labeled $\times$ is possible. Otherwise, the propagation is impossible.

Table 8.5: The possible propagation of the division property for FUTURE Sbox.

	v															
u	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
1		×	×	×		×	×	×	×	×	×	×	×	×	×	×
2		×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
3		×		×		×	×	×	×	×	×	×	×	×	×	×
4		×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
5		×	×	×		×	×	×	×	×	×	×	×	×	×	×
6			×	×		×	×	×		×	×	×		×	×	×
7								×		×	×	×		×	×	×
8		×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
9		×	×	×		×	×	×	×	×	×	×	×	×	×	×
a		×		×	×	×	×	×	×	×	×	×	×	×	×	×
b		×		×		×	×	×	×	×	×	×	×	×	×	×
c			×	×		×	×	×			×	×	×	×	×	×
d			×	×		×	×	×			×	×		×	×	×
e											×	×		×	×	×
f																×

Taking into account the effect of MixColumn, we evaluated the propagation of the division property on the reduced-round FUTURE. To search for the longest integral distinguisher, we choose only one bit in plaintext as constant and the others are active. For example, in the 6th round we have a distinguisher $ACA^{62} \rightarrow B^{64}$. But we could not find any distinguisher for the 7th round by MILP model due to its long running time. So we cannot conclude whether there is an integral distinguisher in the 7th round or not. We also checked that there is no distinguisher from the 8th round [7]. So we are expecting that full rounds of FUTURE is secure against integral attack.

8.4.5 Invariant Subspace Attacks

The invariant subspace attack [LAAZ11, LMR15] exploits a subspace A and constants $u,\ v$ such that $F(u \oplus A) = v \oplus A$, where F is a round transformation of a block cipher. For the round key $r_k \in A \oplus u \oplus v$, $F \oplus r_k$ maps the subspace $u \oplus A$ onto itself, because $F(u \oplus A) \oplus r_k = v \oplus A \oplus r_k = u \oplus A$. However, we can avoid this invariant subspace by using appropriate round constants.

[7] For finding an r round division trail $(a_0^0, a_1^0, \ldots, a_{63}^0) \rightarrow \ldots \rightarrow (a_0^r, a_1^r, \ldots, a_{63}^r)$ by the MILP technique, we fixed the output $(a_0^r, a_1^r, \ldots, a_{63}^r)$ of the rth round by the unit vectors (64 cases) and check whether the MILP model is feasible or not.

By Section 3.3 of [LR18], if RC be the constants on a single cell over all rounds, then the designer can choose RC such that there is no 2-dimensional subspace V of $\mathbb{F}_{2^4}$ satisfying $RC \subseteq V$ for the resistance of invariant subspace attack on AES-like ciphers with MDS MixColumn layer.

Recall that FUTURE is an AES-like ciphers with MDS MixColumn layer which uses round constants 0, 1, 2, 4, 8 in each output of a cell. Also, there is no 2-dimensional subspace V such $\{0, 1, 2, 4, 8\} \subseteq V$. Hence, in FUTURE, the invariant subspace attack cannot be found for an arbitrary number of rounds.

8.4.6 Meet-in-the-Middle Attacks

This section shows the security of FUTURE against the meet-in-the-middle attacks. We have used an approach which is similar to the methods used in the block ciphers MIDORI [BBI+15] and SKINNY [BJK+16]. The maximum number of rounds that can be attacked can be evaluated by considering the maximum length of three features: partial-matching, initial structure, and splice-and-cut.

a. **Partial-matching:** Partial-matching cannot work if the number of rounds reaches full diffusion in each of the forward and backward directions. In FUTURE, full diffusion is achieved after 2 rounds forwards and backwards. Thus, the number of rounds used for partial-matching is upper bounded by $(2-1)+(2-1)+1=3$.

b. **Initial structure:** The condition for the initial structure is that key differential trails in the forward direction and those in the backward direction do not share active Sboxes. For FUTURE, since any key differential affects all 16 Sboxes after at least 4 rounds in the forward and the backward directions, there is no such differential which shares active Sbox in more than 4 rounds. Thus, the number of rounds used for the initial structure is upper bounded by $(4-1)=3$.

c. **Splice-and-cut:** Splice-and-cut may extend the number of attack rounds up to the smaller number of full diffusion rounds minus one, which is $(2-1)=1$.

Therefore, we can conclude that the meet-in-the-middle attack may work up to $3+3+1=$ 7 rounds. Hence, full round FUTURE is sufficient to resist meet-in-the-middle attacks.

8.4.7 Algebraic Attacks

FUTURE Sbox has algebraic degree 3 and from Table 8.4 we see that for 4-round differential characteristic, there are at least 25 active Sboxes. So we have $3 \times 25 \times$

$\lfloor \frac{10}{4} \rfloor = 150 > 64$, where 64 is the block size and 10 is the number of rounds in FUTURE. Also, the FUTURE Sbox is described by 21 quadratic equations in the 8 input/output-bit variables over $\mathbb{F}_2$. The key schedule of FUTURE does not need any Sbox. Thus, the 10-round cipher is described by $10 \times 16 \times 21 = 3360$ quadratic equations in $10 \times 16 \times 8 = 1280$ variables.

The general problem of solving a system of multivariate quadratic equations is NP-hard. However the systems derived for block ciphers are very sparse since they are composed of a small number of nonlinear systems connected by linear layers. Nevertheless, it is unclear whether this fact can be exploited in a so-called algebraic attack. Some specialized techniques such as XL [CKPS00] and XSL [CP02] have been proposed, though flaws in both techniques have been discovered [CL05, Die04]. Instead the practical results on the algebraic cryptanalysis of block ciphers have been obtained by applying the Buchberger and F4 algorithms within Magma. Also, recently there are some practical results [YLK21] on algebraic cryptanalysis by using ElimLin [CB07, CSSV12] and SAT solver techniques [BCJ07, SNC09].

Now note that the entire system for a fixed-key AES permutation consists of 6400 equations in 2560 variables and whereas in FUTURE these numbers are roughly half of that in AES. Simulations on small-scale variants of the AES showed that except for very small versions, one quickly encounters difficulties with time and memory complexity [CMR05]. So we believe that algebraic attacks do not threaten FUTURE.

8.5 Hardware Implementations, Performance and Comparison

In this section, we will discuss the hardware implementation cost of FUTURE in both FPGA and ASIC design.

8.5.1 FPGA Implementation

In recent times, there has been a growing trend of utilizing FPGAs in a wide range of applications, including security and cryptographic domains, due to their high performance capabilities. Given the availability of numerous FPGA vendors, we made the decision to implement our designs on different FPGA boards provided by Xilinx. The hardware implementation of FUTURE is written in VHDL and is implemented on both Virtex-6 and Virtex-7. More specifically, the FPGA results are obtained after place-and-route (PAR) on the Xilinx Virtex-6 (xc6vlx240t-2ff1156)

and Virtex-7 (xc7vx415t-2ffg1157) in Xilinx ISE. In Table 8.6 the implementation results are given. Note that for the comparison of FUTURE with other block ciphers (in fully unrolled implementations), we used the VHDL codes available at https://github.com/KULeuven-COSIC/UnrolledBlockCiphers.

Table 8.6: Results are obtained after PAR for Virtex-6 and Virtex-7.

Cipher	**Virtex-6**			**Virtex-7**		
	Size (Slices)	Critical Path (ns)	Throughput ($Gbit/s$)	Size (Slices)	Critical Path (ns)	Throughput ($Gbit/s$)
KATAN 64/80	2550	47.33	1.35	2550	42.11	1.52
PRESENT 64/80	2089	29.21	2.19	2089	26.27	2.44
PRESENT 64/128	2203	32.55	1.97	2203	29.03	2.20
SIMON 64/128	2688	27.31	2.34	2688	25.30	2.53
SPECK 64/128	3594	50.29	1.27	3594	48.31	1.32
PRINCE	1244	16.38	3.91	1244	14.79	4.33
FUTURE	**1240**	**15.94**	**4.01**	**1241**	**14.53**	**4.40**

8.5.2 ASIC implementation

In order to estimate the hardware cost for an ASIC platform, we will consider the use of the Synopsys Design Compiler using the UMCL18G212T3 [UMC04] ASIC standard cell library, i.e. UMC 0.18µm. In Table 8.7 we describe the area requirements and corresponding gate count in this library (for details, check [Pos09]). Also, note that Gate equivalent (GE) is a measure of the area requirements of integrated circuits (IC). It is derived by dividing the area of the IC by the area of a two-input NAND gate with the lowest driving strength.

However, as mentioned in [SIH+11], certain libraries offer specialized gates that offer additional area savings. Specifically, in this library, 4-input AND-NOR and 4-input OR-NAND gates with two inverted inputs can be utilized to directly compute XOR or XNOR operations. Since both cells cost 2 GE instead of 2.67 GE required for XOR or XNOR, we can save 0.67 GE per XOR or XNOR gate. Now we will discuss

Table 8.7: Area requirements and corresponding gate count.

Standard cell	**Area in μm^2**	**GE**
NAND	9.677	1
NOR	9.677	1
AND/OR	12.902	1.33
XOR/XNOR	25.805	2.67
NOT	6.451	0.67

the cost for each module of a single round FUTURE using the above mentioned implementation techniques.

Cost of FUTURE Sbox: Recall that the FUTURE Sbox S is formed by composing four Sboxes S_1, S_2, S_3, and S_4. Specifically, the Sbox S can be represented as $S(x) = S_1 \circ S_2 \circ S_3 \circ S_4(x)$. The algebraic normal forms of these individual Sboxes are as follows:

$$S_4 : \begin{aligned} y_3 &= x_3 \\ y_2 &= x_1x_3 \oplus x_2 \\ y_1 &= x_1 \\ y_0 &= x_0 \end{aligned} \qquad S_3 : \begin{aligned} y_3 &= x_0x_2 \oplus x_3 \\ y_2 &= x_2 \\ y_1 &= x_1 \\ y_0 &= x_0 \end{aligned}$$

$$S_2 : \begin{aligned} y_3 &= x_3 \\ y_2 &= x_2 \\ y_1 &= x_0 \\ y_0 &= x_0x_3 \oplus x_1 \oplus 1 \end{aligned} \qquad S_1 : \begin{aligned} y_3 &= x_3 \\ y_2 &= x_2 \\ y_1 &= x_0x_2 \oplus x_1 \\ y_0 &= x_0 \end{aligned}$$

Here $y_3y_2y_1y_0$ and $x_3x_2x_1x_0$ denotes the 4-bit output and input respectively of the Sboxes.

From Figures 8-4, 8-5, 8-6 and 8-7, we can observe that the implementation of the FUTURE Sbox requires 4 NAND gates, 3 XNOR gates, and 1 XOR gate. Consequently, FUTURE Sbox can be implemented with $(4 \times 1 + 3 \times 2 + 1 \times 2) = 12$ GE.

Therefore, SubCell operation for a single round FUTURE takes $16 \times 12 = 192$ GE for implementation.

Figure 8-4: Sbox S_4.

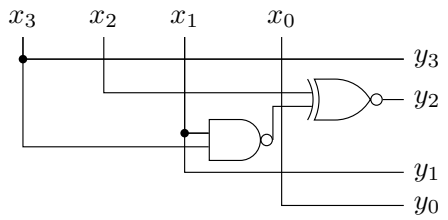


Figure 8-5: Sbox S_3.

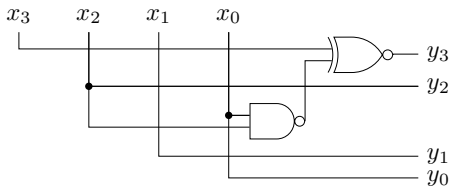


Figure 8-6: Sbox S_2.

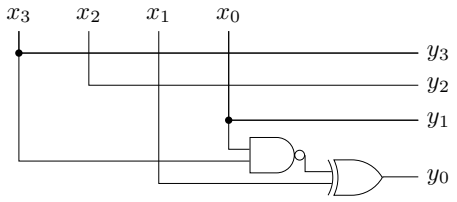
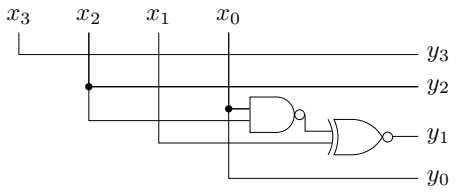


Figure 8-7: Sbox S_1.



Cost of FUTURE MDS matrix: The MDS matrix in FUTURE is a composition of 4 different lightweight sparse matrices M_1, M_2, M_3 and M_4 (see 8.1 of Section 8.2.1).

These matrices are of the form

$$\begin{bmatrix} 0 & 0 & m_1 & m_2 \\ m_3 & 0 & 0 & 0 \\ m_4 & m_5 & 0 & 0 \\ 0 & 0 & m_6 & 0 \end{bmatrix} = GDLS(\rho_1, \rho_2; D_1, D_2),$$

where $\rho_1 = [2, 3, 4, 1]$, $\rho_2 = [3, 2, 1, 4]$, $D_1 = diag(m_3, m_5, m_6, m_2)$, $D_2 = diag(m_4, 0, m_1, 0)$ and $m_i \in \mathbb{F}_{2^4}$ for $1 \leq i \leq 6$. Also, each of the matrices has fixed XOR $\mathcal{K} = 2$ (as discussed in Section 2.6.1). Therefore, for $1 \leq i \leq 4$, we have

$$\begin{aligned} XOR(M_i) &= \sum_{s,t=1}^{4} XOR((M_i)_{s,t}) + 2 \cdot 4 \\ &= \sum_{s=1}^{6} XOR(m_s) + 8. \end{aligned}$$

Thus, the cost for implementation of the matrices are given below:

(a) cost for implementing $M_4 = 8$ XORs.

(b) cost for implementing $M_3 = 1 + 8 = 9$ XORs (the multiplication cost of $\alpha^3 + 1$ is 1 XOR).

(c) cost for implementing $M_2 = 1 + 1 + 8 = 10$ XORs (the multiplication cost of $\alpha^3 + 1$ and α is 1 XOR).

(d) cost for implementing $M_1 = 8$ XORs.

Therefore, MDS matrix for FUTURE needs 35 XOR gates. As a result, it can be implemented with $35 \times 2 =$ 70 GE and MixColumn operation for a single round FUTURE takes $4 \times 70 =$ 280 GE for implementation.

Cost of ShiftRow: Since the ShiftRow operation is essentially a permutation of the entire state, this module can be implemented using a simple wire shuffle and incurs no additional area overhead.

Cost of Key schedule and round constants: Since the round keys are obtained by only bit wiring of the master key, it needs no cost in hardware. Whereas, for the full encryption FUTURE uses 128 NOT gates for the round constants. Therefore, it takes $128 \times 0.67 =$ 85.76 GE. Also the 64-bit round key is XORed with the entire state in each round (also for whitening key) resulting in a $64 \times 2 =$ 128 GE cost for this operation in each single round.

Cost for the full encryption of FUTURE: Since FUTURE is implemented in a fully unrolled fashion, it does not need any extra logic and state register. Therefore, we have the details cost estimations of FUTURE below:

(i) cost for one single round= $192 +$ 280 $+$ 128 $= 600$ GE. So for 9 full rounds, it will cost $9 \times 600 =$ 5400 GE.

(ii) cost for the last round= $192 + 128 =$ 320 GE.

(iii) cost for round constant= 85.76 GE and key whitening needs 128 GE.

Thus, FUTURE can be implemented with $5400 +$ 320 $+$ 85.76 $+$ 128 $=$ 5933.76 GE only. Of course, these numbers depend on the library used, but we expect that it will take less area than our estimations.

In Table 8.8, we list the hardware cost of unrolled implementations for FUTURE and compare it to other block ciphers taken from the literature.

The above table contains the cost estimations of FUTURE along with the cost of other ciphers obtained from Table 12 and Table 24 of [BJK+16]. It should be pointed

Table 8.8: Comparison of the hardware cost of unrolled implementations for FUTURE and other 64-bit ciphers with 128 bit key.

Ciphers	**Area (GE)**
LED-64-128	111496
PRESENT-64-128	56722
PICCOLO-64-128	25668
SKINNY-64-128	17454
MANTIS_5	8544
PRINCE	8512
FUTURE	**5934**

out that SKINNY and MANTIS are tweakable block ciphers, whereas the others are not.

It will be inappropriate to compare the hardware cost of the unrolled version of a rolled block cipher with a large number of rounds because the hardware cost of making the rolled version into the unrolled version will be very high. That's why we are not comparing the hardware cost of FUTURE with the recent block ciphers like GIFT [BPP+17] and CRAFT [BLMR19].

In Table 8.6, we compare FUTURE with some block ciphers in the FPGA platform and Table 8.8 compares its hardware cost with some block ciphers in the ASIC platform. A better approach would be to compare our block cipher with other block ciphers in both FPGA and ASIC implementations. But we are comparing some block ciphers in FPGA and other block ciphers in ASIC because of the unavailability of their hardware codes in the literature.

8.6 Conclusion

One of the fundamental primitives for cryptographic applications is block ciphers. In this chapter, we have proposed a new SPN-based lightweight block cipher, FUTURE, that is designed for minimal latency with low hardware implementation cost. For the best diffusion in the linear layer, it employs an MDS matrix in the round function. Whereas, due to the high cost of MDS matrices, most lightweight block ciphers do not use such matrices in their round function. FUTURE tackles the issue by strategically selecting a highly efficient MDS matrix, which is composed of four sparse matrices. Additionally, FUTURE employs a lightweight yet cryptographically significant Sbox, which is a composition of four different Sboxes. Also, FUTURE shows its resistance

to fundamental attacks. Therefore, by incorporating these design choices, FUTURE successfully combines lightweight implementation with the desirable properties of MDS matrices, offering an effective solution for designing lightweight block ciphers.

Bibliography

[ABI+18] Gianira N. Alfarano, Christof Beierle, Takanori Isobe, Stefan Kölbl, and Gregor Leander. Shiftrows Alternatives for AES-like Ciphers and Optimal Cell Permutations for Midori and Skinny. *IACR Transactions on Symmetric Cryptology*, 2018(2):20–47, Jun. 2018.

[ADK+14] Martin R. Albrecht, Benedikt Driessen, Elif Bilge Kavun, Gregor Leander, Christof Paar, and Tolga Yalçın. Block Ciphers – Focus on the Linear Layer (feat. PRIDE). In Juan A. Garay and Rosario Gennaro, editors, *Advances in Cryptology – CRYPTO 2014*, pages 57–76, Berlin, Heidelberg, 2014. Springer Berlin Heidelberg.

[AF15] Daniel Augot and Matthieu Finiasz. Direct Construction of Recursive MDS Diffusion Layers Using Shortened BCH Codes. In Carlos Cid and Christian Rechberger, editors, *Fast Software Encryption*, pages 3–17, Berlin, Heidelberg, 2015. Springer Berlin Heidelberg.

[Ava17] Roberto Avanzi. The QARMA Block Cipher Family. Almost MDS Matrices Over Rings With Zero Divisors, Nearly Symmetric Even-Mansour Constructions With Non-Involutory Central Rounds, and Search Heuristics for Low-Latency S-Boxes. *IACR Transactions on Symmetric Cryptology*, 2017(1):4–44, Mar. 2017.

[BBI+15] Subhadeep Banik, Andrey Bogdanov, Takanori Isobe, Kyoji Shibutani, Harunaga Hiwatari, Toru Akishita, and Francesco Regazzoni. Midori: A Block Cipher for Low Energy. In Tetsu Iwata and Jung Hee Cheon,

editors, *Advances in Cryptology – ASIACRYPT 2015*, pages 411–436, Berlin, Heidelberg, 2015. Springer Berlin Heidelberg.

[BBK+13] Begül Bilgin, Andrey Bogdanov, Miroslav Knežević, Florian Mendel, and Qingju Wang. Fides: Lightweight Authenticated cipher with Side-Channel Resistance for Constrained Hardware. In Guido Bertoni and Jean-Sébastien Coron, editors, *Cryptographic Hardware and Embedded Systems - CHES 2013*, pages 142–158, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[BBS99] Eli Biham, Alex Biryukov, and Adi Shamir. Cryptanalysis of Skipjack Reduced to 31 Rounds Using Impossible Differentials. In Jacques Stern, editor, *Advances in Cryptology — EUROCRYPT '99*, pages 12–23, Berlin, Heidelberg, 1999. Springer Berlin Heidelberg.

[BBS05] Eli Biham, Alex Biryukov, and Adi Shamir. Cryptanalysis of Skipjack Reduced to 31 Rounds Using Impossible Differentials. *Journal of Cryptology*, 18(4):291–311, Sep 2005.

[BBS+21] Christof Beierle, Alex Biryukov, Luan Cardoso Dos Santos, Johann Großschädl, Léo Perrin, Aein Rezaei Shahmirzadi, Aleksei Udovenko, Vesselin Velichkov, and Qingju Wang. SCHWAEMM and ESCH: Lightweight Authenticated Encryption and Hashing using the Sparkle Permutation Family v1.2, submission to the nist lightweight cryptography competition. Technical report, 2021. https://csrc.nist.gov/Projects/lightweight-cryptography/finalists.

[BCD+21] Zhenzhen Bao, Avik Chakraborti, Nilanjan Datta, Jian Guo, Mridul Nandi, Thomas Peyrin, and Kan Yasuda. PHOTON-Beetle Authenticated Encryption and Hash Family, submission to the nist lightweight cryptography competition. Technical report, 2021. https://csrc.nist.gov/Projects/lightweight-cryptography/finalists.

[BCDM21] Tim Beyne, Yu Long Chen, Christoph Dobraunig, and Bart Mennink. Elephant v2, submission to the nist lightweight cryptography competition. Technical report, 2021. https://csrc.nist.gov/Projects/lightweight-cryptography/finalists.

[BCG+12] Julia Borghoff, Anne Canteaut, Tim Güneysu, Elif Bilge Kavun, Miroslav Knezevic, Lars R. Knudsen, Gregor Leander, Ventzislav Nikov,

Christof Paar, Christian Rechberger, Peter Rombouts, Søren S. Thomsen, and Tolga Yalçın. PRINCE – A Low-Latency Block Cipher for Pervasive Computing Applications. In Xiaoyun Wang and Kazue Sako, editors, *Advances in Cryptology – ASIACRYPT 2012*, pages 208–225, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[BCI+21] Subhadeep Banik, Avik Chakraborti, Tetsu Iwata, Kazuhiko Minematsu, Mridul Nandi, Thomas Peyrin, Yu Sasaki, Siang Meng Sim, and Yosuke Todo. GIFT-COFB v1.1, submission to the nist lightweight cryptography competition. Technical report, 2021. https://csrc.nist.gov/Projects/lightweight-cryptography/finalists.

[BCJ07] Gregory V. Bard, Nicolas T. Courtois, and Chris Jefferson. Efficient Methods for Conversion and Solution of Sparse Systems of Low-Degree Multivariate Polynomials over GF(2) via SAT-Solvers. Cryptology ePrint Archive, Report 2007/024, 2007. https://ia.cr/2007/024.

[BCP97] Wieb Bosma, John Cannon, and Catherine Playoust. The MAGMA Algebra System I: The User Language. *Journal of Symbolic Computation*, 24(3–4):235–265, 1997.

[BDH+17] Guido Bertoni, Joan Daemen, Seth Hoffert, Michaël Peeters, Gilles Van Assche, and Ronny Van Keer. Farfalle: parallel permutation-based cryptography. *IACR Transactions on Symmetric Cryptology*, 2017(4):1–38, Dec. 2017.

[BDPA07] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. *Sponge Functions*. 01 2007.

[BDPA11] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. *The KECCAK reference*. January 2011. http://keccak.noekeon.org/.

[BDPVA10] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. Sponge-Based Pseudo-Random Number Generators. In Stefan Mangard and François-Xavier Standaert, editors, *Cryptographic Hardware and Embedded Systems, CHES 2010*, pages 33–47, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.

[Bei18] Christof Beierle. *Design and Analysis of Lightweight Block Ciphers: a Focus on the Linear Layer, PhD Thesis*. Ruhr University Bochum,

Germany, 2018. https://hss-opus.ub.ruhr-uni-bochum.de/opus4/frontdoor/index/index/docId/5634.

[Ber13] Thierry P. Berger. Construction of Recursive MDS Diffusion Layers from Gabidulin codes. In Goutam Paul and Serge Vaudenay, editors, *Progress in Cryptology – INDOCRYPT 2013*, pages 274–285, Cham, 2013. Springer International Publishing.

[BFI19] Subhadeep Banik, Yuki Funabiki, and Takanori Isobe. More Results on Shortest Linear Programs. In Nuttapong Attrapadung and Takeshi Yagi, editors, *Advances in Information and Computer Security*, pages 109–128, Cham, 2019. Springer International Publishing.

[BJK+16] Christof Beierle, Jérémy Jean, Stefan Kölbl, Gregor Leander, Amir Moradi, Thomas Peyrin, Yu Sasaki, Pascal Sasdrich, and Siang Meng Sim. The SKINNY Family of Block Ciphers and Its Low-Latency Variant MANTIS. In Matthew Robshaw and Jonathan Katz, editors, *Advances in Cryptology – CRYPTO 2016*, pages 123–153, Berlin, Heidelberg, 2016. Springer Berlin Heidelberg.

[BKL+07] Andrey Bogdanov, Lars R. Knudsen, Gregor Leander, Christof Paar, Axel Poschmann, Matthew J. B. Robshaw, Yannick Seurin, and C. Vikkelsoe. PRESENT: An Ultra-Lightweight Block Cipher. In Pascal Paillier and Ingrid Verbauwhede, editors, *Cryptographic Hardware and Embedded Systems - CHES 2007*, pages 450–466, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg.

[BKL16] Christof Beierle, Thorsten Kranz, and Gregor Leander. Lightweight Multiplication in GF(2^n) with Applications to MDS matrices. In Matthew Robshaw and Jonathan Katz, editors, *Advances in Cryptology – CRYPTO 2016*, pages 625–653, Berlin, Heidelberg, 2016. Springer Berlin Heidelberg.

[BL16] Karthikeyan Bhargavan and Gaëtan Leurent. On the Practical (In-)Security of 64-bit Block Ciphers: Collision Attacks on HTTP over TLS and OpenVPN. CCS '16, page 456–467, New York, NY, USA, 2016. Association for Computing Machinery.

[BLMR19] Christof Beierle, Gregor Leander, Amir Moradi, and Shahram Rasoolzadeh. CRAFT: Lightweight Tweakable Block Cipher with Efficient

Protection Against DFA Attacks. *IACR Transactions on Symmetric Cryptology*, 2019(1):5–45, Mar. 2019.

[BMP12] Joan Boyar, Philip Matthews, and René Peralta. Logic Minimization Techniques with Applications to Cryptology. *Journal of Cryptology*, 26:280–312, 2012.

[BO04] Thierry Pierre Berger and Alexei Ourivski. Construction of New MDS Codes from Gabidulin Codes. In *ACCT'9*, Kranevo, Bulgarie., pages 40–47, France, 2004. Springer-Verlag.

[BP17] Alex Biryukov and Leo Perrin. State of the art in lightweight symmetric cryptography. Cryptology ePrint Archive, Paper 2017/511, 2017. https://eprint.iacr.org/2017/511.

[BPP+17] Subhadeep Banik, Sumit Kumar Pandey, Thomas Peyrin, Yu Sasaki, Siang Meng Sim, and Yosuke Todo. GIFT: A Small Present - Towards Reaching the Limit of Lightweight Encryption. In Wieland Fischer and Naofumi Homma, editors, *Cryptographic Hardware and Embedded Systems – CHES 2017*, pages 321–345. Springer International Publishing, 2017.

[BR00a] Paulo S.L.M. Barreto and Vincent Rijmen. The ANUBIS Block Cipher. *Submission to the NESSIE Project, 2000.*, 2000.

[BR00b] Paulo S.L.M. Barreto and Vincent Rijmen. The KHAZAD Legacy-Level Block Cipher. *Submission to the NESSIE Project, 2000.*, 2000.

[BR00c] Paulo S.L.M. Barreto and Vincent Rijmen. The WHIRLPOOL Hashing Function. *Submission to the NESSIE Project, 2000.*, 2000.

[BR22] Tim Beyne and Vincent Rijmen. Differential Cryptanalysis in the Fixed-key Model. Cryptology ePrint Archive, Paper 2022/837, 2022. https://eprint.iacr.org/2022/837.

[BS91a] Eli Biham and Adi Shamir. Differential Cryptanalysis of DES-like Cryptosystems. In Alfred J. Menezes and Scott A. Vanstone, editors, *Advances in Cryptology-CRYPTO' 90*, pages 2–21, Berlin, Heidelberg, 1991. Springer Berlin Heidelberg.

[BS91b] Eli Biham and Adi Shamir. Differential cryptanalysis of DES-like cryptosystems. *Journal of Cryptology*, 4(1):3–72, 1991. https://doi.org/10.1007/BF00630563.

[BSS+15] Ray Beaulieu, Douglas Shors, Jason Smith, Stefan Treatman-Clark, Bryan Weeks, and Louis Wingers. The SIMON and SPECK Lightweight Block Ciphers. DAC '15, New York, NY, USA, 2015. Association for Computing Machinery.

[Car21] Claude Carlet. *Boolean Functions for Cryptography and Coding Theory*. Cambridge University Press, 2021.

[CB07] Nicolas T. Courtois and Gregory V. Bard. Algebraic Cryptanalysis of the Data Encryption Standard. In Steven D. Galbraith, editor, *Cryptography and Coding*, pages 152–169, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg.

[CCJ+16] Tingting Cui, Shiyao Chen, Keting Jia, Kai Fu, and Meiqin Wang. New Automatic Search Tool for Impossible Differentials and Zero-Correlation Linear Approximations. Cryptology ePrint Archive, Report 2016/689, 2016. https://ia.cr/2016/689.

[CJK15] Ting Cui, Chenhui Jin, and Zhiyin Kong. On Compact Cauchy Matrices for Substitution-Permutation Networks. *IEEE Transactions on Computers*, 64(7):2098–2102, 2015.

[CK08] Jiali Choy and Khoongming Khoo. New Applications of Differential Bounds of the SDS Structure. In Tzong-Chen Wu, Chin-Laung Lei, Vincent Rijmen, and Der-Tsai Lee, editors, *Information Security*, pages 367–384, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.

[CKPS00] Nicolas T. Courtois, Alexander Klimov, Jacques Patarin, and Adi Shamir. Efficient Algorithms for Solving Overdefined Systems of Multivariate Polynomial Equations. In Bart Preneel, editor, *Advances in Cryptology — EUROCRYPT 2000*, pages 392–407, Berlin, Heidelberg, 2000. Springer Berlin Heidelberg.

[CL05] Carlos Cid and Gaëtan Leurent. An Analysis of the XSL Algorithm. In Bimal Roy, editor, *Advances in Cryptology - ASIACRYPT 2005*, pages 333–352, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.

[CL19] Victor Cauchois and Pierre Loidreau. On circulant involutory MDS matrices. *Designs, Codes and Cryptography*, 87:149–260, 2019.

[CMR05] Carlos Cid, Sean Murphy, and Matthew J. B. Robshaw. Small Scale Variants of the AES. In Henri Gilbert and Helena Handschuh, editors, *Fast Software Encryption*, pages 145–162, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.

[CMSvS91] G. Castagnoli, J.L. Massey, P.A. Schoeller, and N. von Seemann. On repeated-root cyclic codes. *IEEE Transactions on Information Theory*, 37(2):337–342, 1991.

[CP02] Nicolas T. Courtois and Josef Pieprzyk. Cryptanalysis of Block Ciphers with Overdefined Systems of Equations. In Yuliang Zheng, editor, *Advances in Cryptology — ASIACRYPT 2002*, pages 267–287, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg.

[CSSV12] Nicolas T. Courtois, Pouyan Sepehrdad, Petr Sušil, and Serge Vaudenay. Elimlin Algorithm Revisited. In Anne Canteaut, editor, *Fast Software Encryption*, pages 306–325, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[CYK+12] Jiali Choy, Huihui Yap, Khoongming Khoo, Jian Guo, Thomas Peyrin, Axel Poschmann, and Chik How Tan. SPN-Hash: Improving the Provable Resistance against Differential Collision Attacks. In Aikaterini Mitrokotsa and Serge Vaudenay, editors, *Progress in Cryptology - AFRICACRYPT 2012*, pages 270–286, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[Dae95] Joan Daemen. *Cipher and hash function design, strategies based on linear and differential cryptanalysis, PhD Thesis*. K.U.Leuven, 1995. http://jda.noekeon.org/.

[DB96] Mario A De Boer. Almost MDS codes. 9(2):143–155, October 1996.

[DCDK09] Christophe De Cannière, Orr Dunkelman, and Miroslav Knežević. KATAN and KTANTAN — A Family of Small and Efficient Hardware-Oriented Block Ciphers. In Christophe Clavier and Kris Gaj, editors, *Cryptographic Hardware and Embedded Systems - CHES 2009*, pages 272–288, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[DEM+21] Christoph Dobraunig, Maria Eichlseder, Stefan Mangard, Florian Mendel, Bart Mennink, Robert Primas, and Thomas Unterluggauer. Isap v2.0, submission to the nist lightweight cryptography competition. Technical report, 2021. https://csrc.nist.gov/Projects/lightweight-cryptography/finalists.

[DEMS21] Christoph Dobraunig, Maria Eichlseder, Florian Mendel, and Martin Schläffer. ASCON v1.2, submission to the nist lightweight cryptography competition. Technical report, 2021. https://csrc.nist.gov/Projects/lightweight-cryptography/finalists.

[DGV91] Joan Daemen, René Govaerts, and Joos Vandewalle. A framework for the design of one-way hash functions including cryptanalysis of damgård's one-way function based on a cellular automaton. In Hideki Imai, Ronald L. Rivest, and Tsutomu Matsumoto, editors, *Advances in Cryptology — ASIACRYPT '91*, pages 82–96, Berlin, Heidelberg, 1991. Springer Berlin Heidelberg.

[DGV93] Joan Daemen, René Govaerts, and Joos Vandewalle. A new Approach to Block Cipher Design. In Ross Anderson, editor, *Fast Software Encryption*, pages 18–32, Berlin, Heidelberg, 1993. Springer Berlin Heidelberg.

[DGV95] Joan Daemen, René Govaerts, and Joos Vandewalle. Correlation Matrices. In Bart Preneel, editor, *Fast Software Encryption*, pages 275–285, Berlin, Heidelberg, 1995. Springer Berlin Heidelberg.

[DHAK18] Joan Daemen, Seth Hoffert, Gilles Van Assche, and Ronny Van Keer. The design of Xoodoo and Xoofff. *IACR Transactions on Symmetric Cryptology*, 2018(4):1–38, Dec. 2018.

[DHM+21] Joan Daemen, Seth Hoffert, Silvia Mella, Michaël Peeters, Gilles Van Assche, and Ronny Van Keer. Xoodyak, a lightweight cryptographic scheme v2, submission to the nist lightweight cryptography competition. Technical report, 2021. https://csrc.nist.gov/Projects/lightweight-cryptography/finalists.

[Die04] Claus Diem. The XL-Algorithm and a Conjecture from Commutative Algebra. In Pil Joong Lee, editor, *Advances in Cryptology - ASIACRYPT 2004*, pages 323–337, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg.

[DKR97] Joan Daemen, Lars Knudsen, and Vincent Rijmen. The block cipher Square. In Eli Biham, editor, *Fast Software Encryption*, pages 149–165, Berlin, Heidelberg, 1997. Springer Berlin Heidelberg.

[DL95] Stefan Dodunekov and Ivan Landgev. On near-MDS codes. *Journal of Geometry*, 54(1):30–43, 1995.

[DL18] Sébastien Duval and Gaëtan Leurent. MDS Matrices with Lightweight Circuits. *IACR Transactions on Symmetric Cryptology*, 2018(2):48–78, Jun. 2018.

[DPAR00] Joan Daemen, Michaël Peeters, Gilles Van Assche, and Vincent Rijmen. Nessie Proposal: NOEKEON. In *First Open NESSIE Workshop*, 2000.

[DR99] Joan Daemen and Vincent Rijmen. AES Proposal: Rijndael. Submission to the NIST AES competition, 1999.

[DR02] Joan Daemen and Vincent Rijmen. *The Design of Rijndael: AES - The Advanced Encryption Standard*. Information Security and Cryptography. Springer, 2002.

[Dwo01] Morris Dworkin. Recommendation for Block Cipher Modes of Operation: Methods and Techniques. Technical report, NIST Special Publication 800-38A, 2001. https://doi.org/10.6028/NIST.SP.800-38A.

[Eic18] Maria Eichlseder. *Differential Cryptanalysis of Symmetric Primitives, PhD Thesis*. Graz University of Technology, Austria, 2018. https://graz.elsevierpure.com/en/publications/differential-cryptanalysis-of-symmetric-primitives.

[EM03] Moawwad E.A. El-Mikkawy. Explicit inverse of a generalized Vandermonde matrix. *Applied Mathematics and Computation*, 146(2):643–651, 2003.

[FBR06] Décio Luiz Gazzoni Filho, Paulo S.L.M. Barreto, and Vincent Rijmen. The Maelstrom-0 hash function. *In Proceedings of the 6th Brazilian Symposium on Information and Computer Systems Security*, 2006.

[FNS75] HORST Feistel, WILLIAM A. Notz, and J. LYNN Smith. Some cryptographic techniques for machine-to-machine data communications. *Proceedings of the IEEE*, 63(11):1545–1554, 1975.

[GIK+21] Chun Guo, Tetsu Iwata, Mustafa Khairallah, Kazuhiko Minematsu, and Thomas Peyrin. Romulus v1.3, submission to the nist lightweight cryptography competition. Technical report, 2021. https://csrc.nist.gov/Projects/lightweight-cryptography/finalists.

[GKM+08] Praveen Gauravaram, Lars Knudsen, Krystian Matusiewicz, Florian Mendel, Christian Rechberger, Martin Schläffer, and Søren Thomsen. Grøstl - a SHA-3 candidate. *Submission to NIST, 2008, Available at http://www.groestl.info/*, 09 2008.

[GKPS04] DARYL GELLER, Irwin Kra, SORIN POPESCU, and Santiago Simanca. On Circulant Matrices. *Notices of the American Mathematical Society*, 59, 01 2004.

[GKR78] I Gohberg, M.A Kaashoek, and L Rodman. Spectral analysis of families of operator polynomials and a generalized Vandermonde matrix ii: The infinite dimensional case. *Journal of Functional Analysis*, 30(3):358–389, 1978.

[GPP11] Jian Guo, Thomas Peyrin, and Axel Poschmann. The PHOTON Family of Lightweight Hash Functions. In Phillip Rogaway, editor, *Advances in Cryptology – CRYPTO 2011*, pages 222–239, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[GPPR11] Jian Guo, Thomas Peyrin, Axel Poschmann, and Matt Robshaw. The LED Block Cipher. In Bart Preneel and Tsuyoshi Takagi, editors, *Cryptographic Hardware and Embedded Systems – CHES 2011*, pages 326–341, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[GPRS19] Kishan Chand Gupta, Sumit Kumar Pandey, Indranil Ghosh Ray, and Susanta Samanta. Cryptographically significant MDS matrices over finite fields: A brief survey and some generalized results. *Advances in Mathematics of Communications*, 13(4):779–843, 2019.

[GPS19] Kishan Chand Gupta, Sumit Kumar Pandey, and Susanta Samanta. A Few Negative Results on Constructions of MDS Matrices Using Low XOR Matrices. In Shivam Bhasin, Avi Mendelson, and Mridul Nandi, editors, *Security, Privacy, and Applied Cryptography Engineering*, pages 195–213, Cham, 2019. Springer International Publishing.

[GPS22a] Kishan Chand Gupta, Sumit Kumar Pandey, and Susanta Samanta. Construction of Recursive MDS Matrices Using DLS Matrices. In Lejla Batina and Joan Daemen, editors, *Progress in Cryptology - AFRICACRYPT 2022*, pages 3–27, Cham, 2022. Springer Nature Switzerland.

[GPS22b] Kishan Chand Gupta, Sumit Kumar Pandey, and Susanta Samanta. FUTURE: A Lightweight Block Cipher Using an Optimal Diffusion Matrix. In Lejla Batina and Joan Daemen, editors, *Progress in Cryptology - AFRICACRYPT 2022*, pages 28–52, Cham, 2022. Springer Nature Switzerland.

[GPS23a] Kishan Chand Gupta, Sumit Kumar Pandey, and Susanta Samanta. On the construction of near-MDS matrices. *Cryptography and Communications*, Aug 2023.

[GPS23b] Kishan Chand Gupta, Sumit Kumar Pandey, and Susanta Samanta. On the Direct Construction of MDS and Near-MDS Matrices. arXiv:2306.12848, 2023. https://arxiv.org/abs/2306.12848.

[GPV15] Kishan Chand Gupta, Sumit Kumar Pandey, and Ayineedi Venkateswarlu. Towards a General Construction of Recursive MDS Diffusion Layers. In Pascale Charpin, Nicolas Sendrier, and Jean-Pierre Tillich, editors, *The 9th International Workshop on Coding and Cryptography 2015 WCC2015*, Proceedings of the 9th International Workshop on Coding and Cryptography 2015 WCC2015, Paris, France, April 2015.

[GPV17a] Kishan Chand Gupta, Sumit Kumar Pandey, and Ayineedi Venkateswarlu. On the direct construction of recursive MDS matrices. *Designs, Codes and Cryptography*, 82(1-2):77–94, 2017.

[GPV17b] Kishan Chand Gupta, Sumit Kumar Pandey, and Ayineedi Venkateswarlu. Towards a general construction of recursive MDS diffusion layers. *Designs, Codes and Cryptography*, 82(1-2):179–195, 2017.

[GPV19] Kishan Chand Gupta, Sumit Kumar Pandey, and Ayineedi Venkateswarlu. Almost involutory recursive MDS diffusion layers. *Designs, Codes and Cryptography*, 87(2-3):609–626, 2019.

[GR13a] Kishan Chand Gupta and Indranil Ghosh Ray. On Constructions of Involutory MDS Matrices. In Amr Youssef, Abderrahmane Nitaj, and Aboul Ella Hassanien, editors, *Progress in Cryptology – AFRICACRYPT 2013*, pages 43–60, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[GR13b] Kishan Chand Gupta and Indranil Ghosh Ray. On Constructions of MDS Matrices from Companion Matrices for Lightweight Cryptography. In Alfredo Cuzzocrea, Christian Kittl, Dimitris E. Simos, Edgar Weippl, and Lida Xu, editors, *Security Engineering and Intelligence Informatics*, pages 29–43, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[GR14] Kishan Chand Gupta and Indranil Ghosh Ray. On Constructions of Circulant MDS Matrices for Lightweight Cryptography. In Xinyi Huang and Jianying Zhou, editors, *Information Security Practice and Experience*, pages 564–576, Cham, 2014. Springer International Publishing.

[GR15] Kishan Chand Gupta and Indranil Ghosh Ray. Cryptographically significant MDS matrices based on circulant and circulant-like matrices for lightweight applications. *Cryptography and Communications*, 7:257–287, 2015.

[Gra06] Robert M. Gray. Toeplitz and Circulant Matrices: A Review. *Foundations and Trends in Communications and Information Theory*, 2(3):155–239, 2006.

[Hey02] Howard M. Heys. A Tutorial on Linear and Differential Cryptanalysis. 26(3):189–221, Jul 2002. https://doi.org/10.1080/0161-110291890885.

[Hir95] J. W. P. Hirschfeld. The main conjecture for MDS codes. In Colin Boyd, editor, *Cryptography and Coding*, pages 44–52, Berlin, Heidelberg, 1995. Springer Berlin Heidelberg.

[HJM+21] Martin Hell, Thomas Johansson, Alexander Maximov, Willi Meier, Jonathan Sönnerup, and Hirotaka Yoshida. Grain-128AEAD v2- A lightweight AEAD stream cipher, submission to the nist lightweight cryptography competition. Technical report, 2021. https://csrc.nist.gov/Projects/lightweight-cryptography/finalists.

[HT94] Howard M. Heys and Stafford E. Tavares. The Design of Substitution-Permutation Networks Resistant to Differential and Linear Cryptanalysis. In *Proceedings of the 2nd ACM Conference on Computer and Communications Security*, CCS '94, pages 148–155, New York, NY, USA, 1994. Association for Computing Machinery.

[HT95] Howard M. Heys and Stafford E. Tavares. Avalanche characteristics of substitution-permutation encryption networks. *IEEE Transactions on Computers*, 44(9):1131–1139, 1995.

[HT96] Howard M. Heys and Stafford E. Tavares. Substitution-permutation networks resistant to differential and linear cryptanalysis. *Journal of Cryptology*, 9:1–19, 1996.

[HYNL21] Daitao Huang, Qin Yue, Yongfeng Niu, and Xia Li. MDS or NMDS self-dual codes from twisted generalized Reed-Solomon codes. *Designs, Codes and Cryptography*, 89(9):2195–2209, Sep 2021.

[JM09] Pascal Junod and Marco Macchetti. Revisiting the IDEA Philosophy. In Orr Dunkelman, editor, *Fast Software Encryption*, pages 277–295, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[JPST17] Jérémy Jean, Thomas Peyrin, Siang Meng Sim, and Jade Tourteaux. Optimizing Implementations of Lightweight Building Blocks. *IACR Transactions on Symmetric Cryptology*, 2017(4):130–168, Dec. 2017.

[JV05a] Pascal Junod and Serge Vaudenay. FOX: A New Family of Block Ciphers. In Helena Handschuh and M. Anwar Hasan, editors, *Selected Areas in Cryptography*, pages 114–129, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.

[JV05b] Pascal Junod and Serge Vaudenay. Perfect Diffusion Primitives for Block Ciphers. In Helena Handschuh and M. Anwar Hasan, editors, *Selected Areas in Cryptography*, pages 84–99, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.

[KLK09] Nicholas Kolokotronis, Konstantinos Limniotis, and Nicholas Kalouptsidis. Factorization of determinants over finite fields and application in stream ciphers. *Cryptography and Communications*, 1:175–205, 2009.

[KLSW17] Thorsten Kranz, Gregor Leander, Ko Stoffelen, and Friedrich Wiemer. Shorter Linear Straight-Line Programs for MDS Matrices. *IACR Transactions on Symmetric Cryptology*, 2017(4):188–211, Dec. 2017.

[Knu94] Lars Ramkilde Knudsen. *Block Ciphers: Analysis, Design and Applications, PhD Thesis*. Aarhus University, Denmark, 1994. https://tidsskrift.dk/daimipb/article/view/6978.

[Knu98] Lars Knudsen. DEAL–A 128-bit Block Cipher. *NIST AES Proposal*, 1998.

[Köl19] Lukas Kölsch. XOR-Counts and Lightweight Multiplication with Fixed Elements in Binary Finite Fields. In Yuval Ishai and Vincent Rijmen, editors, *Advances in Cryptology – EUROCRYPT 2019*, pages 285–312, Cham, 2019. Springer International Publishing.

[KPPY14] Khoongming Khoo, Thomas Peyrin, Axel Y. Poschmann, and Huihui Yap. FOAM: Searching for Hardware-Optimal SPN Structures and Components with a Fair Comparison. In Lejla Batina and Matthew Robshaw, editors, *Cryptographic Hardware and Embedded Systems – CHES 2014*, pages 433–450, Berlin, Heidelberg, 2014. Springer Berlin Heidelberg.

[KPSV21] Abhishek Kesarwani, Sumit Kumar Pandey, Santanu Sarkar, and Ayineedi Venkateswarlu. Recursive MDS matrices over finite commutative rings. *Discret. Appl. Math.*, 304:384–396, 2021.

[KSV19] Abhishek Kesarwani, Santanu Sarkar, and Ayineedi Venkateswarlu. Exhaustive Search for Various Types of MDS Matrices. *IACR Transactions on Symmetric Cryptology*, 2019(3):231–256, Sep. 2019.

[LAAZ11] Gregor Leander, Mohamed Ahmed Abdelraheem, Hoda AlKhzaimi, and Erik Zenner. A Cryptanalysis of PRINTcipher: The Invariant Subspace Attack. In Phillip Rogaway, editor, *Advances in Cryptology – CRYPTO 2011*, pages 206–221, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[LF04a] Jérôme Lacan and Jérôme Fimes. A Construction of Matrices with No Singular Square Submatrices. In Gary L. Mullen, Alain Poli, and Henning Stichtenoth, editors, *Finite Fields and Applications*, pages 145–147, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg.

[LF04b] Jérôme Lacan and Jérôme Fimes. Systematic MDS erasure codes based on Vandermonde matrices. *IEEE Communications Letters*, 8(9):570–572, 2004.

[LMM91] Xuejia Lai, James L. Massey, and Sean Murphy. Markov Ciphers and Differential Cryptanalysis. In Donald W. Davies, editor, *Advances in Cryptology — EUROCRYPT '91*, pages 17–38, Berlin, Heidelberg, 1991. Springer Berlin Heidelberg.

[LMR15] Gregor Leander, Brice Minaud, and Sondre Rønjom. A Generic Approach to Invariant Subspace Attacks: Cryptanalysis of Robin, iSCREAM and Zorro. In Elisabeth Oswald and Marc Fischlin, editors, *Advances in Cryptology – EUROCRYPT 2015*, pages 254–283, Berlin, Heidelberg, 2015. Springer Berlin Heidelberg.

[LN97] Rudolf Lidl and Harald Niederreiter. *Finite Fields*. Encyclopedia of Mathematics and its Applications. Cambridge University Press, 2 edition, 1997.

[LR18] Yunwen Liu and Vincent Rijmen. New observations on invariant subspace attack. *Information Processing Letters*, 138:27–30, 2018.

[LRW02] Moses Liskov, Ronald L. Rivest, and David Wagner. Tweakable Block Ciphers. In Moti Yung, editor, *Advances in Cryptology — CRYPTO 2002*, pages 31–46, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg.

[LRW11] Moses Liskov, Ronald L Rivest, and David Wagner. Tweakable Block Ciphers. *Journal of Cryptology*, 24(3):588–613, July 2011.

[LS16] Meicheng Liu and Siang Meng Sim. Lightweight MDS Generalized Circulant Matrices. In Thomas Peyrin, editor, *Fast Software Encryption*, pages 101–120, Berlin, Heidelberg, 2016. Springer Berlin Heidelberg.

[LSL+19] Shun Li, Siwei Sun, Chaoyun Li, Zihao Wei, and Lei Hu. Constructing Low-latency Involutory MDS Matrices with Lightweight Circuits. *IACR Transactions on Symmetric Cryptology*, 2019(1):84–117, Mar. 2019.

[LSS+20] Shun Li, Siwei Sun, Danping Shi, Chaoyun Li, and Lei Hu. Lightweight Iterative MDS Matrices: How Small Can We Go? *IACR Transactions on Symmetric Cryptology*, 2019(4):147–170, Jan 2020.

[LW17] Chaoyun Li and Qingju Wang. Design of lightweight linear diffusion layers from near-mds matrices. *IACR Transactions on Symmetric Cryptology*, 2017(1):129–155, Mar. 2017.

[LW21] Xiaodan Li and Wenling Wu. Constructions of Iterative Near-MDS Matrices with the Lowest XOR Count. In Joonsang Baek and Sushmita Ruj, editors, *Information Security and Privacy*, pages 132–150, Cham, 2021. Springer International Publishing.

[Mat94] Mitsuru Matsui. Linear Cryptanalysis Method for DES Cipher. In Tor Helleseth, editor, *Advances in Cryptology — EUROCRYPT '93*, pages 386–397, Berlin, Heidelberg, 1994. Springer Berlin Heidelberg.

[MRS12] Ferdaouss Mattoussi, Vincent Roca, and Bessem Sayadi. Complexity comparison of the use of Vandermonde versus Hankel matrices to build systematic MDS Reed-Solomon codes. In *2012 IEEE 13th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pages 344–348, 2012.

[MRV15] Bart Mennink, Reza Reyhanitabar, and Damian Vizár. Security of Full-State Keyed Sponge and Duplex: Applications to Authenticated Encryption. In Tetsu Iwata and Jung Hee Cheon, editors, *Advances in Cryptology – ASIACRYPT 2015*, pages 465–489, Berlin, Heidelberg, 2015. Springer Berlin Heidelberg.

[MS77] F.J. MacWilliams and N.J.A. Sloane. *The Theory of Error Correcting Codes*. North-Holland Publishing Co., Amsterdam-New York-Oxford, 1977.

[NA09] Jorge Nakahara and Élcio Abrahão. A New Involutory MDS Matrix for the AES. *International Journal of Network Security*, 9:109–116, 2009.

[Nyb95] Kaisa Nyberg. Linear Approximation of Block Ciphers. In Alfredo De Santis, editor, *Advances in Cryptology — EUROCRYPT'94*, pages 439–444, Berlin, Heidelberg, 1995. Springer Berlin Heidelberg.

[Ota22] Kamil Otal. A Generalization of the Subfield Construction. *International Journal of Information Security Science*, 11(2):1–11, 2022.

[Pos09] Axel Poschmann. Lightweight Cryptography - Cryptographic Engineering for a Pervasive World. Cryptology ePrint Archive, Report 2009/516, 2009. https://ia.cr/2009/516.

[Pow67] Henry M. Power. The companion matrix and Liapunov functions for linear multivariable time-invariant systems. *Journal of the Franklin Institute*, 283(3):214–234, 1967.

[PSA+18] Meltem Kurt PehlIvanoğlu, Muharrem Tolga Sakallı, Sedat Akleylek, Nevcihan Duru, and Vincent Rijmen. Generalisation of Hadamard matrix to generate involutory MDS matrices for lightweight cryptography. *IET Information Security*, 12(4):348–355, 2018.

[PSAS22] Meltem Kurt PehlIvanoğlu, Fatma Büyüksaraçoglu Sakalli, Sedat Akleylek, and Muharrem Tolga Sakalli. On the Construction of 4×4 Lightweight Involutory MDS Matrices Over $\mathbb{F}_{2^8}$. In Debasis Giri, Kim-Kwang Raymond Choo, Saminathan Ponnusamy, Weizhi Meng, Sedat Akleylek, and Santi Prasad Maity, editors, *Proceedings of the Seventh International Conference on Mathematics and Computing*, volume 1412 of *Advances in Intelligent Systems and Computing*, pages 725–736. Springer, 2022.

[PSAS23] Meltem Kurt PehlIvanoğlu, Fatma Büyüksaraçoglu Sakalli, Sedat Akleylek, and Muharrem Tolga Sakalli. On the Construction of New Lightweight Involutory MDS Matrices in Generalized Subfield Form. *IEEE Access*, 11:32708–32715, 2023.

[PUB77] PUB. FIPS. 46: Data encryption standard (DES). National Institute of Standards and Technology, 1977. http://csrc.nist.gov/publications/fips/fips46-3/fips46-3.pdf.

[PUB01] PUB. FIPS. 197: Advanced encryption standard (AES). National Institute of Standards and Technology, 2001. http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf.

[PUB15] PUB. *FIPS. 180-4: Secure Hash Standard - SHS.* U.S. Department of Commerce and National Institute of Standards and Technology, North Charleston, SC, USA, 2015. https://doi.org/10.6028/NIST.FIPS.180-4.

[RB00] A. Ramachandra Rao and P. Bhimasankaram. *Linear Algebra*. Hindustan Book Agency, 2000.

[RDP+96] Vincent Rijmen, Joan Daemen, Bart Preneel, Antoon Bosselaers, and Erik De Win. The cipher SHARK. In Dieter Gollmann, editor, *Fast Software Encryption*, pages 99–111, Berlin, Heidelberg, 1996. Springer Berlin Heidelberg.

[RL89] Ron M. Roth and Abraham Lempel. On MDS codes via Cauchy matrices. *IEEE Transactions on Information Theory*, 35(6):1314–1319, 1989.

[Rog11] Phillip Rogaway. *Evaluation of Some Blockcipher Modes of Operation*. CRYPTREC Report, 2011.

[Rot76] O.S Rothaus. On "bent" functions. *Journal of Combinatorial Theory, Series A*, 20(3):300–305, 1976.

[RS85] Ron M. Roth and Gadiel Seroussi. On generator matrices of MDS codes. *IEEE Transactions on Information Theory*, 31(6):826–830, 1985.

[RSA78] R. L. Rivest, A. Shamir, and L. Adleman. A Method for Obtaining Digital Signatures and Public-key Cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.

[RT09] Fred S. Roberts and Barry Tesman. *Applied Combinatorics, Second Edition*. CRC Press; Chapman and Hall/CRC, 2009.

[SAAR20] Muharrem Tolga Sakalli, Sedat Akleylek, Kemal Akkanat, and Vincent Rijmen. On the automorphisms and isomorphisms of MDS matrices and their efficient implementations. *Turkish Journal of Electrical Engineering and Computer Sciences*, 28(1):275–287, 2020.

[Sch98] Bruce Schneier. The Twofish Encryption Algorithm. *Dr. Dobb's Journal: Software Tools for the Professional Programmer*, 23(12):30–34, 1998.

[SDMO12] Mahdi Sajadieh, Mohammad Dakhilalian, Hamid Mala, and Behnaz Omoomi. On construction of Involutory MDS Matrices from Vandermonde Matrices in $GF(2^q)$. *Designs, Codes and Cryptography*, 64(3):287–308, sep 2012.

[SDMS12] Mahdi Sajadieh, Mohammad Dakhilalian, Hamid Mala, and Pouyan Sepehrdad. Recursive Diffusion Layers for Block Ciphers and Hash functions. In Anne Canteaut, editor, *Fast Software Encryption*, pages 385–401, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[Sha49] C. E. Shannon. Communication theory of secrecy systems. *The Bell System Technical Journal*, 28(4):656–715, 1949.

[Shp05] Igor E. Shparlinski. On the singularity of generalised Vandermonde matrices over finite fields. *Finite Fields and Their Applications*, 11(2):193–199, 2005.

[SIH+11] Kyoji Shibutani, Takanori Isobe, Harunaga Hiwatari, Atsushi Mitsuda, Toru Akishita, and Taizo Shirai. Piccolo: An Ultra-Lightweight Blockcipher. In Bart Preneel and Tsuyoshi Takagi, editors, *Cryptographic Hardware and Embedded Systems – CHES 2011*, pages 342–357, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[SKOP15] Siang Meng Sim, Khoongming Khoo, Frédérique Oggier, and Thomas Peyrin. Lightweight MDS Involution Matrices. In Gregor Leander, editor, *Fast Software Encryption*, pages 471–493, Berlin, Heidelberg, 2015. Springer Berlin Heidelberg.

[SKW+98] Bruce Schneier, John Kelsey, Doug Whiting, David Wagner, Chris Hall, and Niels Ferguson. Twofish: A 128-Bit Block Cipher. In *In First Advanced Encryption Standard (AES) Conference*, 1998.

[SM21] Mahdi Sajadieh and Mohsen Mousavi. Construction of MDS matrices from generalized feistel structures. *Designs, Codes and Cryptography*, 89:1433–1452, 2021.

[SMMK13] Tomoyasu Suzaki, Kazuhiko Minematsu, Sumio Morioka, and Eita Kobayashi. TWINE: A Lightweight Block Cipher for Multiple Platforms. In Lars R. Knudsen and Huapeng Wu, editors, *Selected Areas in Cryptography*, pages 339–354, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[SNC09] Mate Soos, Karsten Nohl, and Claude Castelluccia. Extending SAT Solvers to Cryptographic Problems. In Oliver Kullmann, editor, *Theory*

and Applications of Satisfiability Testing - SAT 2009, pages 244–257, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[SS03] Taizo Shirai and Kyoji Shibutani. On the diffusion matrix employed in the Whirlpool hashing function. *NESSIE public report, 2003*, 2003.

[SS16] Sumanta Sarkar and Habeeb Syed. Lightweight Diffusion Layer: Importance of Toeplitz Matrices. *IACR Transactions on Symmetric Cryptology*, 2016(1):95–113, Dec. 2016.

[SS17] Sumanta Sarkar and Habeeb Syed. Analysis of Toeplitz MDS Matrices. In Josef Pieprzyk and Suriadi Suriadi, editors, *Information Security and Privacy*, pages 3–18, Cham, 2017. Springer International Publishing.

[SSA+07] Taizo Shirai, Kyoji Shibutani, Toru Akishita, Shiho Moriai, and Tetsu Iwata. The 128-bit Blockcipher CLEFIA (Extended Abstract). In Alex Biryukov, editor, *Fast Software Encryption*, pages 181–195, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg.

[SSL15] Kazuo Sakiyama, Yu Sasaki, and Yang Li. *Security of Block Ciphers: From Algorithm Design to Hardware Implementation*. Wiley Publishing, 1st edition, 2015.

[ST17] Yu Sasaki and Yosuke Todo. New Impossible Differential Search Tool from Design and Cryptanalysis Aspects. In Jean-Sébastien Coron and Jesper Buus Nielsen, editors, *Advances in Cryptology – EUROCRYPT 2017*, pages 185–215. Springer International Publishing, 2017.

[SV95] C. P. Schnorr and S. Vaudenay. Black box cryptanalysis of hash networks based on multipermutations. In Alfredo De Santis, editor, *Advances in Cryptology — EUROCRYPT'94*, pages 47–57, Berlin, Heidelberg, 1995. Springer Berlin Heidelberg.

[SWW20] Ling Sun, Wei Wang, and Meiqin Q. Wang. MILP-aided bit-based division property for primitives with non-bit-permutation linear layers. *IET Information Security*, 14(1):12–20, 2020.

[SYLH22] Junzhen Sui, Qin Yue, Xia Li, and Daitao Huang. MDS, Near-MDS or 2-MDS Self-Dual Codes via Twisted Generalized Reed-Solomon Codes. *IEEE Transactions on Information Theory*, 68(12):7832–7841, 2022.

[SZZ94] Jennifer Seberry, Xian-Mo Zhang, and Yuliang Zheng. Relationships Among Nonlinear Criteria (Extended Abstract). In *Advances in Cryptology - EUROCRYPT '94, Workshop on the Theory and Application of Cryptographic Techniques, Perugia, Italy, May 9-12, 1994, Proceedings*, pages 376–388, 1994.

[TCG92] Anne Tardy-Corfdir and Henri Gilbert. A Known Plaintext Attack of FEAL-4 and FEAL-6. In Joan Feigenbaum, editor, *Advances in Cryptology — CRYPTO '91*, pages 172–182, Berlin, Heidelberg, 1992. Springer Berlin Heidelberg.

[TM16] Yosuke Todo and Masakatu Morii. Bit-Based Division Property and Application to Simon Family. In Thomas Peyrin, editor, *Fast Software Encryption*, pages 357–377, Berlin, Heidelberg, 2016. Springer Berlin Heidelberg.

[TMC+23] Meltem Sönmez Turan, Kerry McKay, Donghoon Chang, Lawrence E. Bassham, Jinkeon Kang, Noah D. Waller, John M. Kelsey, and Deukjo Hong. Status Report on the Final Round of the NIST Lightweight Cryptography Standardization Process. NIST Interagency or Internal Report (IR) NIST IR 8454, National Institute of Standards and Technology, Gaithersburg, MD, 2023. https://doi.org/10.6028/NIST.IR.8454.

[TSP+23] Gökhan Tuncay, Fatma Büyüksaraçoglu Sakalli, Meltem Kurt PehlIvanoğlu, Gülsüm Gözde Yilmazgüç, Sedat Akleylek, and Muharrem Tolga Sakalli. A new hybrid method combining search and direct based construction ideas to generate all 4×4 involutory maximum distance separable (MDS) matrices over binary field extensions. *PeerJ Comput. Sci.*, 9:e1577, 2023.

[TTKS18] Dylan Toh, Jacob Teo, Khoongming Khoo, and Siang Meng Sim. Lightweight MDS Serial-Type Matrices with Minimal Fixed XOR count. In Antoine Joux, Abderrahmane Nitaj, and Tajjeeddine Rachidi, editors, *Progress in Cryptology – AFRICACRYPT 2018*, pages 51–71, Cham, 2018. Springer International Publishing.

[UMC04] UMCL18G212T3. Virtual Silicon Inc. 0.18 μm VIP Standard Cell Library Tape Out Ready, Part Number: UMCL18G212T3, Process: UMC Logic 0.18 μm Generic II Technology: 0.18μm, July 2004.

[Van77] H. Van de Vel. Numerical treatment of a generalized Vandermonde system of equations. *Linear Algebra and its Applications*, 17(2):149–179, 1977.

[Vau95] Serge Vaudenay. On the need for multipermutations: Cryptanalysis of MD4 and SAFER. In Bart Preneel, editor, *Fast Software Encryption*, pages 286–297, Berlin, Heidelberg, 1995. Springer Berlin Heidelberg.

[VL91] J.H. Van Lint. Repeated-root cyclic codes. *IEEE Transactions on Information Theory*, 37(2):343–345, 1991.

[VR06] G. Viswanath and Balaji Sundar Rajan. A Matrix Characterization of Near-MDS codes. *Ars Comb.*, 79:289–294, 2006.

[Wag99] David Wagner. The Boomerang Attack. In Lars Knudsen, editor, *Fast Software Encryption*, pages 156–170, Berlin, Heidelberg, 1999. Springer Berlin Heidelberg.

[Wei91] V.K. Wei. Generalized hamming weights for linear codes. *IEEE Transactions on Information Theory*, 37(5):1412–1418, 1991.

[WFY+02] Dai Watanabe, Soichi Furuya, Hirotaka Yoshida, Kazuo Takaragi, and Bart Preneel. A New Keystream Generator MUGI. In Joan Daemen and Vincent Rijmen, editors, *Fast Software Encryption*, pages 179–194, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg.

[WH21] Hongjun Wu and Tao Huang. TinyJAMBU: A Family of Lightweight Authenticated Encryption Algorithms v2, submission to the nist lightweight cryptography competition. Technical report, 2021. https://csrc.nist.gov/Projects/lightweight-cryptography/finalists.

[WT86] A. F. Webster and S. E. Tavares. On the Design of S-Boxes. In Hugh C. Williams, editor, *Advances in Cryptology — CRYPTO '85 Proceedings*, pages 523–534, Berlin, Heidelberg, 1986. Springer Berlin Heidelberg.

[WWW13] Shengbao Wu, Mingsheng Wang, and Wenling Wu. Recursive Diffusion Layers for (Lightweight) Block Ciphers and Hash Functions. In Lars R. Knudsen and Huapeng Wu, editors, *Selected Areas in Cryptography*, pages 355–371, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[XTL14] Hong Xu, Lin Tan, and Xuejia Lai. On the Recursive Construction of MDS Matrices for Lightweight Cryptography. In Xinyi Huang and Jianying Zhou, editors, *Information Security Practice and Experience*, pages 552–563, Cham, 2014. Springer International Publishing.

[XZBL16] Zejun Xiang, Wentao Zhang, Zhenzhen Bao, and Dongdai Lin. Applying Milp Method to Searching Integral Distinguishers Based on Division Property for 6 Lightweight Block Ciphers. In Jung Hee Cheon and Tsuyoshi Takagi, editors, *Advances in Cryptology – ASIACRYPT 2016*, pages 648–678, Berlin, Heidelberg, 2016. Springer Berlin Heidelberg.

[YLK21] Sze Ling Yeo, Duc-Phong Le, and Khoongming Khoo. Improved algebraic attacks on lightweight block ciphers. *Journal of Cryptographic Engineering*, 11:1–19, 2021. https://doi.org/10.1007/s13389-020-00237-4.

[YMT97] A. M. Youssef, S. Mister, and Stafford E. Tavares. On the Design of Linear Transformations for Substitution Permutation Encryption Networks. In *Workshop on Selected Areas in Cryptography, SAC*, pages 40–48, 1997.

[YTH96] A. M. Youssef, Stafford E. Tavares, and Howard M. Heys. A New Class of Substitution-Permutation Networks. In *Workshop on Selected Areas in Cryptography, SAC*, pages 132–147, 1996.

[YZW21] Yumeng Yang, Xiangyong Zeng, and Shi Wang. Construction of lightweight involutory MDS matrices. *Designs, Codes and Cryptography*, 89:1–31, 07 2021.

A

Appendix

A.1 Differential Distribution Table of FUTURE Sbox

Table A.1: Differential Distribution Table (DDT) of FUTURE Sbox.

	ΔO															
ΔI	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	4	4	0	0	0	0	0	4	4	0	0	0	0	0
2	0	4	0	4	0	2	0	2	0	0	0	0	2	0	2	0
3	0	0	0	4	2	0	2	0	0	0	4	0	0	2	0	2
4	0	0	0	0	4	0	4	0	0	0	0	0	0	4	0	4
5	0	0	0	0	2	2	2	2	0	0	0	0	2	2	2	2
6	0	4	0	4	0	2	0	2	0	0	0	0	2	0	2	0
7	0	0	4	0	0	2	0	2	0	4	0	0	2	0	2	0
8	0	0	0	0	2	0	2	0	4	2	0	2	4	0	0	0
9	0	2	2	0	0	2	2	0	0	0	2	2	0	0	2	2
a	0	0	0	0	0	4	0	0	4	2	0	2	0	2	0	2
b	0	2	2	0	0	0	2	2	0	0	2	2	2	0	0	2
c	0	0	0	0	2	0	2	0	4	2	0	2	0	0	4	0
d	0	2	2	0	2	0	0	2	0	0	2	2	2	2	0	0
e	0	0	0	0	0	0	0	4	4	2	0	2	0	2	0	2
f	0	2	2	0	2	2	0	0	0	0	2	2	0	2	2	0

A.2 Linear Approximation Table of FUTURE Sbox

Table A.2: Linear Approximation Table (LAT) of FUTURE Sbox. Each entry represents $\#\{x \in \mathbb{F}_{2^4} : x \cdot \alpha \oplus S(x) \cdot \beta = 0\} - 8$.

	β															
α	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	4	4	0	0	4	-4	0	0	0	0	0	0	0	0
2	0	-4	-2	-2	0	0	2	-2	0	-4	2	2	0	0	-2	2
3	0	-4	-2	-2	0	0	2	-2	0	4	-2	-2	0	0	2	-2
4	0	2	0	-2	4	-2	0	-2	2	0	-2	0	2	0	2	4
5	0	-2	4	-2	0	-2	0	2	-2	0	2	0	2	4	2	0
6	0	-2	2	0	4	-2	-2	0	-2	0	0	2	-2	-4	0	-2
7	0	2	2	-4	0	-2	2	0	2	0	0	-2	-2	0	-4	-2
8	0	0	0	0	0	0	0	0	4	0	0	4	4	0	0	-4
9	0	0	0	0	0	0	0	0	4	0	4	0	-4	0	4	0
a	0	0	-2	2	0	-4	2	2	0	-4	-2	-2	0	0	2	-2
b	0	0	2	-2	0	4	-2	-2	0	-4	-2	-2	0	0	2	-2
c	0	-2	0	2	4	2	0	2	2	0	2	-4	2	0	-2	0
d	0	2	0	-2	0	2	4	2	-2	0	2	0	2	-4	2	0
e	0	-2	2	0	-4	-2	-2	0	2	0	0	-2	2	-4	0	2
f	0	2	-2	0	0	-2	-2	-4	-2	0	4	-2	2	0	0	-2

A.3 Test Vectors for FUTURE

Plaintext	Key ($K = K_0 \| \| K_1$)	Ciphertext
0x0000000000000000	0x00000000000000000000000000000000	0x298650c13199cdec
0x0000000000000000	0x00000000000000001111111111111111	0x4aa41b330751b83d
0xffffffffffffff	0x000102030405060708090a0b0c0d0e0f	0x68e030733fe73b8a
0xffffffffffffff	0xffffffffffffffffffffffffffffffff	0x333ba4b7646e09f2
0x6162636465666768	0x00000000000000000000000000000000	0xcc5ba5e52038b6df
0x5353414d414e5441	0x05192832010913645029387763948871	0x5ce1b8d8d01a9310

A.4 T_1 : 4-bit Sboxes implemented by 1 XOR or XNOR gates

Table A.3: 4-bit Sboxes implemented by 1 XOR or XNOR gates (Here $y_3y_2y_1y_0$ and $x_3x_2x_1x_0$ denotes the 4-bit output and input respectively of the Sboxes).

Sbox	ANF
0123456798badcfe	$y_3 = x_3,\ y_2 = x_2,\ y_1 = x_1,\ y_0 = x_0 \oplus x_3$
0123547689abdcfe	$y_3 = x_3,\ y_2 = x_2,\ y_1 = x_1,\ y_0 = x_0 \oplus x_2$
0132457689bacdfe	$y_3 = x_3,\ y_2 = x_2,\ y_1 = x_1,\ y_0 = x_0 \oplus x_1$
1023546798abdcef	$y_3 = x_3,\ y_2 = x_2,\ y_1 = x_1,\ y_0 = x_0 \oplus x_1 \oplus 1$
1032456798bacdef	$y_3 = x_3,\ y_2 = x_2,\ y_1 = x_1,\ y_0 = x_0 \oplus x_2 \oplus 1$
1032547689abcdef	$y_3 = x_3,\ y_2 = x_2,\ y_1 = x_1,\ y_0 = x_0 \oplus x_3 \oplus 1$
01234567ab89efcd	$y_3 = x_3,\ y_2 = x_2,\ y_1 = x_1 \oplus x_3,\ y_0 = x_0$
0123674589abefcd	$y_3 = x_3,\ y_2 = x_2,\ y_1 = x_1 \oplus x_2,\ y_0 = x_0$
031247568b9acfde	$y_3 = x_3,\ y_2 = x_2,\ y_1 = x_0,\ y_0 = x_0 \oplus x_1$
120356479a8bdecf	$y_3 = x_3,\ y_2 = x_2,\ y_1 = x_0,\ y_0 = x_0 \oplus x_1 \oplus 1$
130246579b8acedf	$y_3 = x_3,\ y_2 = x_2,\ y_1 = x_0,\ y_0 = x_1 \oplus x_2 \oplus 1$
130257468a9bcedf	$y_3 = x_3,\ y_2 = x_2,\ y_1 = x_0,\ y_0 = x_1 \oplus x_3 \oplus 1$
01234567cdef89ab	$y_3 = x_3,\ y_2 = x_2 \oplus x_3,\ y_1 = x_1,\ y_0 = x_0$
0167234589efabcd	$y_3 = x_3,\ y_2 = x_1,\ y_1 = x_1 \oplus x_2,\ y_0 = x_0$
034712568bcf9ade	$y_3 = x_3,\ y_2 = x_1,\ y_1 = x_0,\ y_0 = x_0 \oplus x_2$
125603479ade8bcf	$y_3 = x_3,\ y_2 = x_1,\ y_1 = x_0,\ y_0 = x_0 \oplus x_2 \oplus 1$
134602579bce8adf	$y_3 = x_3,\ y_2 = x_1,\ y_1 = x_0,\ y_0 = x_1 \oplus x_2 \oplus 1$
135702468ace9bdf	$y_3 = x_3,\ y_2 = x_1,\ y_1 = x_0,\ y_0 = x_2 \oplus x_3 \oplus 1$
0123cdef456789ab	$y_3 = x_2,\ y_2 = x_2 \oplus x_3,\ y_1 = x_1,\ y_0 = x_0$
016789ef2345abcd	$y_3 = x_2,\ y_2 = x_1,\ y_1 = x_1 \oplus x_3,\ y_0 = x_0$
03478bcf12569ade	$y_3 = x_2,\ y_2 = x_1,\ y_1 = x_0,\ y_0 = x_0 \oplus x_3$
12569ade03478bcf	$y_3 = x_2,\ y_2 = x_1,\ y_1 = x_0,\ y_0 = x_0 \oplus x_3 \oplus 1$
13469bce02578adf	$y_3 = x_2,\ y_2 = x_1,\ y_1 = x_0,\ y_0 = x_1 \oplus x_3 \oplus 1$
13578ace02469bdf	$y_3 = x_2,\ y_2 = x_1,\ y_1 = x_0,\ y_0 = x_2 \oplus x_3 \oplus 1$

A.5 T_2 : 4-bit Sboxes implemented by 1 NAND and 1 XOR/XNOR gates

Table A.4: 4-bit Sboxes implemented by 1 NAND and 1 XOR/XNOR gates (Here $y_3y_2y_1y_0$ and $x_3x_2x_1x_0$ denotes the 4-bit output and input respectively of the Sboxes).

Sbox	ANF
0123456789abdcfe	$y_3 = x_3,\ y_2 = x_2,\ y_1 = x_1,\ y_0 = x_0 \oplus x_2x_3$ [$x_0 \oplus x_2x_3$=XNOR$((x_0, \text{NAND}(x_2, x_3)))$]
0123456789bacdfe	$y_3 = x_3,\ y_2 = x_2,\ y_1 = x_1,\ y_0 = x_0 \oplus x_1x_3$
0123457689abcdfe	$y_3 = x_3,\ y_2 = x_2,\ y_1 = x_1,\ y_0 = x_0 \oplus x_1x_2$
1032546798badcef	$y_3 = x_3,\ y_2 = x_2,\ y_1 = x_1,\ y_0 = x_0 \oplus x_1x_2 \oplus 1$
1032547698abdcef	$y_3 = x_3,\ y_2 = x_2,\ y_1 = x_1,\ y_0 = x_0 \oplus x_1x_3 \oplus 1$ [$x_0 \oplus x_1x_3 \oplus 1$=XOR$((x_0, \text{NAND}(x_1, x_3)))$]
1032547698bacdef	$y_3 = x_3,\ y_2 = x_2,\ y_1 = x_1,\ y_0 = x_0 \oplus x_2x_3 \oplus 1$
0123456789abefcd	$y_3 = x_3,\ y_2 = x_2,\ y_1 = x_1 \oplus x_2x_3,\ y_0 = x_0$
012345678ba9cfed	$y_3 = x_3,\ y_2 = x_2,\ y_1 = x_0x_3 \oplus x_1,\ y_0 = x_0$
0123476589abcfed	$y_3 = x_3,\ y_2 = x_2,\ y_1 = x_0x_2 \oplus x_1,\ y_0 = x_0$
130256479b8adecf	$y_3 = x_3,\ y_2 = x_2,\ y_1 = x_0,\ y_0 = x_0x_2 \oplus x_1 \oplus 1$
130257469a8bdecf	$y_3 = x_3,\ y_2 = x_2,\ y_1 = x_0,\ y_0 = x_0x_3 \oplus x_1 \oplus 1$
130257469b8acedf	$y_3 = x_3,\ y_2 = x_2,\ y_1 = x_0,\ y_0 = x_1 \oplus x_2x_3 \oplus 1$
0123456789efcdab	$y_3 = x_3,\ y_2 = x_1x_3 \oplus x_2,\ y_1 = x_1,\ y_0 = x_0$
012345678dafc9eb	$y_3 = x_3,\ y_2 = x_0x_3 \oplus x_2,\ y_1 = x_1,\ y_0 = x_0$
0127456389afcdeb	$y_3 = x_3,\ y_2 = x_0x_1 \oplus x_2,\ y_1 = x_1,\ y_0 = x_0$
135602479bde8acf	$y_3 = x_3,\ y_2 = x_1,\ y_1 = x_0,\ y_0 = x_0x_1 \oplus x_2 \oplus 1$
135702469ade8bcf	$y_3 = x_3,\ y_2 = x_1,\ y_1 = x_0,\ y_0 = x_0x_3 \oplus x_2 \oplus 1$
135702469bce8adf	$y_3 = x_3,\ y_2 = x_1,\ y_1 = x_0,\ y_0 = x_1x_3 \oplus x_2 \oplus 1$
012345ef89abcd67	$y_3 = x_1x_2 \oplus x_3,\ y_2 = x_2,\ y_1 = x_1,\ y_0 = x_0$
01234d6f89abc5e7	$y_3 = x_0x_2 \oplus x_3,\ y_2 = x_2,\ y_1 = x_1,\ y_0 = x_0$
012789af4563cdeb	$y_3 = x_2,\ y_2 = x_0x_1 \oplus x_3,\ y_1 = x_1,\ y_0 = x_0$
13569bde02478acf	$y_3 = x_2,\ y_2 = x_1,\ y_1 = x_0,\ y_0 = x_0x_1 \oplus x_3 \oplus 1$
13579ade02468bcf	$y_3 = x_2,\ y_2 = x_1,\ y_1 = x_0,\ y_0 = x_0x_2 \oplus x_3 \oplus 1$
13579bce02468adf	$y_3 = x_2,\ y_2 = x_1,\ y_1 = x_0,\ y_0 = x_1x_2 \oplus x_3 \oplus 1$