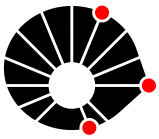

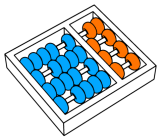

MDS Matrices for Cryptography

T. S. R. Silva *R. Dahab*

Relatório Técnico - IC-PFG-21-43
Projeto Final de Graduação
2021 - Dezembro

UNIVERSIDADE ESTADUAL DE CAMPINAS
INSTITUTO DE COMPUTAÇÃO

MDS Matrices for Cryptography

Tomás S. R. Silva*

Ricardo Dahab†

Abstract

Maximum Distance Separable (MDS) matrices are a key component in several cryptographic schemes. One of the most interesting features, from a cryptographic point of view, of MDS matrices is the fact that these provide perfect diffusion for linear layers.

Thus, this work will not only explore the characteristic of perfect diffusion in MDS layers, but will also demonstrate that the use of MDS matrices is a necessary (but not a sufficient) condition in order to achieve resistance against infinitely long invariant subspace trails attacks in P-SPN linear layers. Moreover, it will also be presented some MDS matrices construction techniques.

Keywords: MDS matrices; P-SPN and SPN; HADES-like cryptosystems; subspace trails attacks.

*Institute of Computing, University of Campinas, Brazil. Email: tomas.silva@students.ic.unicamp.br
†Institute of Computing, University of Campinas, Brazil. Email: rdahab@ic.unicamp.br

Contents

1 Introduction

For thousands of years, Cryptography has been used in order to provide *secure* and *confidential* communication between mutually trusted parties that communicate in an insecure environment (*channel*). To do so, two entities, often denoted as *Alice* and *Bob*, must agree on a particular *secret* to face the presence of an adversary, *Eve*, in the insecure channel, who can monitor all communications between them.

When Alice wants to communicate with Bob, a *key* is agreed, and it is used to feed an *encryption* scheme (*cipher*) to transform the messages in such a way that the adversary cannot distinguish it from random data. The result of a message encryption is called *ciphertext*. When Bob (or Alice) receives a ciphertext, the agreed key is used to transform the ciphertext back into the original *plaintext* in a process called *decryption*. A *cryptosystem* constitutes a complete specification of the keys and how they are used to encrypt and decrypt information.

Various types of cryptosystems of increasing sophistication have been used for many purposes throughout history. Important applications have included sensitive communications between political leaders, royalty, military forces, etc. However, with the development of the internet, many new diverse applications have emerged. These include scenarios such as encryption of passwords, credit card numbers, email, documents, files, and digital media, among others.

The techniques used by an adversary to try to "break" a cryptosystem are named *cryptanalysis*. The most obvious type of cryptanalysis is to try to guess the key agreed between Alice and Bob. An attack in which the adversary tries to decrypt the ciphertext with every possible key in turn is called an *exhaustive key search*. When the adversary tries the correct key, the plaintext will be found, but when any other key is used, the decrypted ciphertext will likely be a random string. So an obvious first step in designing a secure cryptosystem is to specify a very large number of possible keys, so many that the adversary

will not be able to test them all in any reasonable amount of time.

The proper design and implementation of cryptographic primitives thus provides data *integrity*, *authenticity*, *confidentiality*, and *non-repudiation*. Confidentiality consists in ensuring that secret information is available only to authorized parties by means of *encryption*. The encryption process is equivalent to computing a function on sensitive information taking as input an additional token, or key, as well. This function is easy to invert only for those who own a *trapdoor* to this function, which is mathematically related to the key used to compute the encryption. In some cases, the trapdoor can be the key itself. In practice, computing the inverse function is equivalent to solving a computational problem, which is easy to solve only for the trapdoor holder. Otherwise, solving the problem should take exponential time on a conventional computer or even, in some cases, using quantum algorithms.

1.1 Organization of this Work

As will be seen throughout this document, in some criptographic primitives the usage of *Maximum Distance Separable* (MDS) codes play a key role, once they provide perfect *diffusion* for ciphertext and key bits. The remaining work will explore MDS structures focused on cryptographic applications, following the organization shown bellow.

- Section 2 exhibits a brief recapitulation of key terminologies regarding cryptosystems design and security.
- Section 3 shows the main mathematical definitions and results that will be important throughout this work, mostly regarding Linear Algebra and Coding Theory.
- Section 4 discusses some important use cases for MDS matrices in Cryptography, and explains how MDS are well fit for designing the called *non-trivial linear layers*. Particularly, Subsection 4.3.1 shows the main results and proofs proposed in this paper.

- Section 5 presents different construction techniques for MDS matrices that make use of both companion and circulating matrices.
- Section 6 summarizes the major results and discussions presented in this work, and indicates possible topics for future research.

2 Historical Background

2.1 Shannon's Theory

The groundbreaking 1949s paper "*Communication Theory of Secrecy Systems*" published by Claude Shannon [Sha49] had a great influence on the scientific study of cryptography. By considering some elementary probability theory, Shannon introduced many concepts regarding the evaluation of a cryptosystem security.

Computational Security. This concept concerns the computational effort required by an adversary to break a cryptosystem. A cryptosystem might be classified as computationally secure if the best algorithm for breaking it requires at least N operations, where N is some specified large number (usually, N is exponential under some cryptosystem parameter). However, no cryptosystem can be proved to be secure under this definition. Indeed, the computational security of a cryptosystem is usually studied with respect to certain specific types of attacks (e.g., an exhaustive key search), but security against one specific attack does not ensure security against some other type of attack.

Provable Security. Regards the evidence of security by means of a polynomial problem *reduction*. That is, if it is proven that a cryptosystem is as hard to solve as a well known difficult problem (e.g., prime factoring or discrete logarithm), the cryptosystem can be called provably secure. This approach only provides a proof of security relative to some other problem, not an absolute proof of security; similarly to proving that a problem is NP-complete: it proves that the given problem is at least as difficult as any other NP-complete problem, but it does not provide an absolute proof of the computational difficulty.

Unconditional Security. A cryptosystem is defined to be unconditionally secure if it cannot be broken, even with infinite computational resources.

When discussing the security of a cryptosystem, we should also specify the type of attack that is being considered. As will be seen throughout this work, we will be focusing in a called *Invariant Subspace Attack* on block ciphers that rely on MDS codes.

2.2 Block and Stream Ciphers

Cryptosystems are usually categorized as *block ciphers* or *stream ciphers*. In a block cipher, the plaintext is divided into fixed-sized chunks called blocks, where each block will be encrypt or decrypt at a time. As instance, AES is a block cipher with a fixed block size of 128 bits, and a key size of 128, 192, or 256 bits.

In contrast, a stream cipher first uses the key to construct a keystream, which is a bitstring that has exactly the same length as the plaintext, which in turn has an arbitrary length. The encryption operation constructs the ciphertext as the exclusive-or of the plaintext and the keystream. Decryption is accomplished by computing the exclusive-or (XOR) of the ciphertext and the keystream.

Public-key cryptosystems are invariably block ciphers, while secret-key cryptosystems can be block ciphers or stream ciphers.

2.3 Secret-key Cryptosystems

The cryptographic model known as *secrete-key cryptography* (or *symmetric cryptography*) indicates that there is one secret key, which is known to both Alice and Bob. That is, the key is a secret that is known by all the authorized parties. This key is employed both to encrypt plaintexts and to decrypt ciphertexts. The actual encryption and decryption functions are thus inverses of each other.

One of the most common symmetric encryption standards is the Rijndael (AES) cipher [NIS01], proposed in 2001 by Vincent Rijmen and Joan Daemen. AES is based on a design principle

known as a *substitution–permutation network*, and most of its calculations are done in a particular *finite field*.

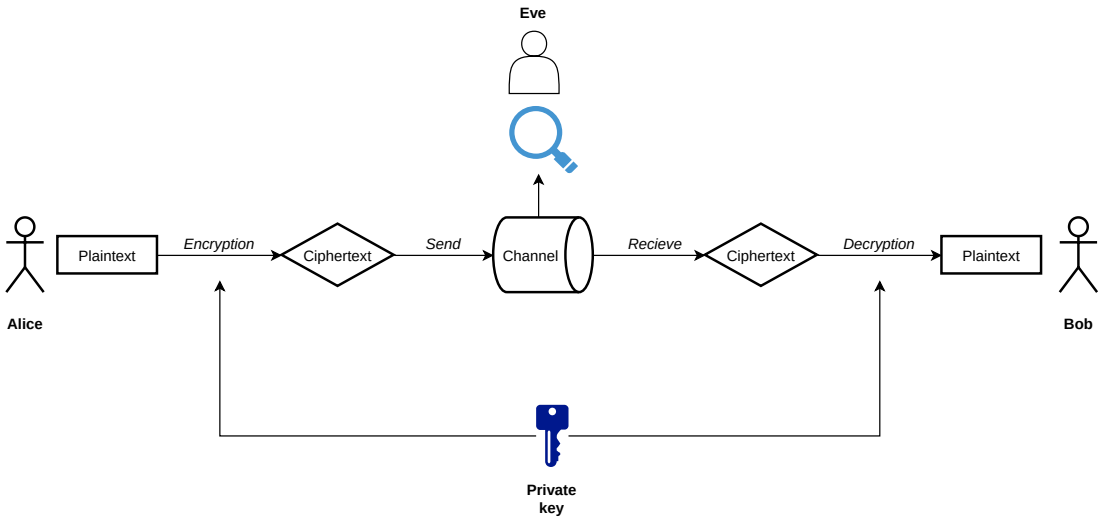

Figure 1: Secret-key cryptosystem schematic.

2.3.1 AES

The *Advanced Encryption Standard* [NIS01] is a standard defined by the US National Institute of Standards and Technology (NIST) for cryptographic protection of electronic data. It was defined in 2001, after a long open standardization process to replace the *Data Encryption Standard* (DES), dated back to the 1970s. It is a standardization of the Rijndael [DR02] family of block ciphers, based on an Substitution Permutation Network. A round consists of the successive application of four steps: `SubBytes`, `ShiftRows`, `MixColumns` and `AddRoundKey`. The first and last round are exceptions, as the `MixColumns` operation is not performed in the last round, and the first round has an additional `AddRoundKey` operation in order to mix the primary key with clear text.

In the algorithm below, r is the current round, R is the total number of rounds, k_0, k_r and k_R are the respective rounds keys.

NIST standardizes amounts R of rounds and lengths n_b in bits for the block to be

Algorithm 2.1 AES

Require: x (plaintext), $k_0, k_1, \ldots, k_R$ (sub-keys)
Ensure: y (ciphertext)
  $x \leftarrow \texttt{AddRoundKey}(x, k_0)$
  for r from 1 to $R-1$ **do**
    $x \leftarrow \texttt{SubBytes}(x)$
    $x \leftarrow \texttt{ShiftRows}(x)$
    $x \leftarrow \texttt{MixColumns}(x)$
    $x \leftarrow \texttt{AddRoundKey}(x, k_r)$
  end for
  $x \leftarrow \texttt{SubBytes}(x)$
  $x \leftarrow \texttt{ShiftRows}(x)$
  $x \leftarrow \texttt{AddRoundKey}(x, k_R)$
  $y \leftarrow x$
  return y

encrypted and the key, thus existing variants AES-128 ($n_b = 128$, $R = 10$), AES-192 ($n_b = 192$, $R = 12$) and AES-256 ($n_b = 256$, $R = 14$). The S-box and other details, such as the procedure for expanding the master key, are detailed in the NIST publication for AES [NIS01].

To better understand what the AES operations `SubBytes`, `ShiftRows`, `MixColumns` and `AddRoundKey` do; the state is represented as a 4x4 matrix where each element is a byte. In this example, we will consider a state of 16 bytes, these being $x_0, x_1, \ldots, x_{15}$. This status is arranged in 4 columns, as follows:

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

SubBytes. Replace each byte of the state according to a S-box ($x_i \leftarrow S(x_i)$). The new

state, after applying a SubBytes, is therefore:

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

ShiftRows. The lines of the matrix go through a cyclic shift, except for the first line. The second line is shifted one cell to the left, the third, two cells to the left, and the fourth, three cells to the left. The result of this will then be:

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

MixColumns. Multiplication of each state column by an involutory circulating MDS matrix $MC = (2, 3, 1, 1)$, that is, each column c_i of the state is replaced by the column $m_i = (MC \times c_i)$. This multiplication is performed by interpreting the bytes that make up the state as polynomials of a particular *finite field* (see Section 3).

$$MC = \begin{bmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 1 \end{bmatrix}$$

AddRoundKey. Addition modulo 2 (or an exclusive or logical operation, i.e. XOR) of each state bit with each corresponding current-round-key bit.

2.4 Public-key Cryptosystems

The cryptographic model known as *public-key cryptography* (or *asymmetric cryptography*) indicates that there is a pair of keys used throughout the cryptosystem.A *public key* would be used to encrypt the plaintext and a *private key* would enable the ciphertext to be decrypted. A public key can be known by anyone (e.g., Eve), whereas a private key is known to only one person, i.e., Alice or Bob.

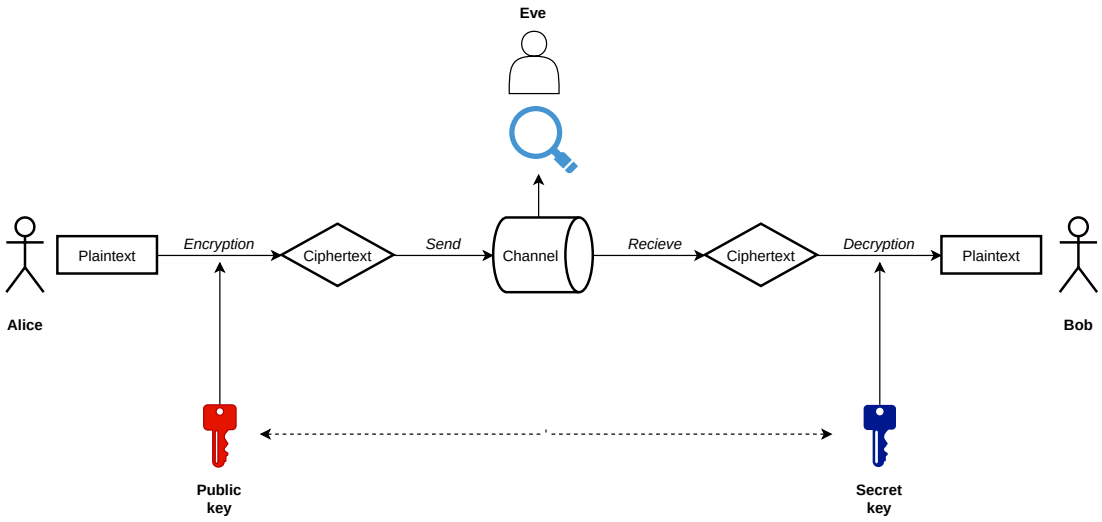

Figure 2: Public-key cryptosystem schematic.

In 1976, Diffie and Hellman [DH76] proposed a secure method for key exchange, and the RSA cryptosystem [RSA78] was published soon after. These and other public-key cryptography systems rely on a one-way trapdoor function, where given x, the function $f(x)$ is easy to compute, but the reverse is computationally hard, unless a secret property t is also given. A *cryptographic hash function* is an example of a one way function where given an input of arbitrary size it produces a fixed size value which is hard to invert. Note however that it is not a trapdoor function, since there is no secret property or value which allows for the reversal of a hash. The textbook version of RSA encodes a message as an integer m and encrypt it by computing $m^e \bmod n$ for an integer e, a key with public access,

and an integer modulus n which defines the message space $\mathbb{Z}_n$. This modulus is the product of two very large prime numbers, both of which remain secret. Computing the inverse function to recover the message m is equivalent to computing the e^{th} root of m^e modulo n, which is hard when the prime factors of n are large enough. However, an authorized party owns an integer d, the trapdoor, for which it holds that $e \cdot d \equiv 1 \bmod \phi(n)$ and d is the only such value mod$\phi(n)$, where $\phi(n)$ is the Euler's totient function. From this property, it holds that decryption is done by simply taking $(m^e)^d = m$. Another way of inverting $m^e \bmod n$ is to factor the modulus n. Finding the exponent d is easy given the factorization of n. However, factoring is also a hard problem when n is the product of two large primes.

3 Mathematical Background

Definition 1 (Groups)**.** *A Group $(G, *)$ is a set G with a binary operation $*$ where*

1. *for all $a, b \in G$, $a * b \in G$*
2. *for all $a, b, c \in G$, $a * (b * c) = (a * b) * c$*
3. *exists an element $e \in G$ such that $a * e = e * a = a$ for all $a \in G$*
4. *for all $a \in G$, exists an element $b \in G$ such that $a * b = b * a = e$*

   Besides that, the Group $(G, *)$ is said Abelian if:

5. *for all $a, b \in G$, $a * b = b * a$*

Definition 2 (Ring)**.** *A Ring $(R, +, .)$ is a set R with two binary operations $+$ and $.$ so that:*

1. *$(R, +)$ is an Abelian Group*
2. *for all $a, b, c \in R$, $a.(b.c) = (a.b).c$*
3. *for all $a, b, c \in R$, $a.(b + c) = a.b + a.c$ and $(b + c).a = b.a + c.a$*

Definition 3 (Field)**.** *A Ring* $(R, +, .)$ *is called Field if* $(R^*, .)$ *is an Abelian Group, where* $R^* = R \setminus \{0\}$

Definition 4 (Finite Field)**.** *If* $(R, +, .)$ *is a Field and the cardinality of* R *is finite,* $|R| = q$*, then we have a Finite Field denoted by* $\mathbb{F}_q$*.*

We can construct vector spaces over finite fields. Let's denote by $\mathbb{F}_q^n$ the n-dimensional vector space constructed with vector entries as elements of $\mathbb{F}_q$.

Thus, we can define linear binary codes and codewords, where any linear combination of its codeword elements also forms a codeword.

Definition 5 (Linear Binary Code)**.** *Let* k, n *be positive integer constants, such that* $k \leq n$*. A linear binary code* $\mathcal{C}[n, k]_2$ *is a vector subspace with dimension* k *of* $\mathbb{F}_2^n$*, (*$\mathcal{C} \subseteq \mathbb{F}_2^n$*). The elements* $c \in \mathcal{C}[n, k]_2$ *are called codewords, and the code size (cardinality) is* $|\mathcal{C}| = 2^k$*.*

The concepts of a generator matrix and a parity check matrix are central in coding theory. A generator matrix can be seen as a matrix where its rows form a basis for a linear code, while a parity check matrix can be derived from a generator matrix of a linear code. The formal definitions are given below.

Definition 6 (Generator Matrix and Parity Check Matrix)**.** *Given a linear code* $\mathcal{C}[n, k]_2$*, we can choose* k *linear independent vectors* $\{g_0, g_1, ..., g_{k-1}\}$ *from* $\mathcal{C}[n, k]_2$*. Each vector of* $\mathcal{C}[n, k]_2$ *can be written as a linear combination of the chosen vectors, i.e.,* $c \in \mathcal{C} \leftrightarrow c = uG$*, where* $u \in \mathbb{F}_2^k$ *is a coordinate vector and* G *is the matrix*

$$G = \begin{bmatrix} g_0 \\ g_1 \\ \vdots \\ g_{k-1} \end{bmatrix},$$

called Generator matrix of $\mathcal{C}$*. This matrix has dimension* $k \times n$*. We can also define the*

matrix $H : r \times k$, $r = n - k$, *such that*

$$Hc^T = \mathbf{0}_{r\times 1},$$

called Parity Check matrix of $\mathcal{C}$.

Definition 7 (Systematic form)**.** *A matrix is said to be systematic if*

$$\mathbf{A} = [\mathbf{I}_m \mid \mathbf{T}]$$

where $\mathbf{I}_m$ *is a* $m \times m$ *identity matrix. For a linear code* $\mathcal{C}$*, its generator or parity matrix in systematic form are both unique. Given a generator or parity matrix, it is possible to obtain the associated systematic form through linear operations such as Gaussian Elimination.*

Given the generator matrix G and a parity check matrix H for a code $\mathcal{C}[n,k]_2$,it is possible to write its systematic form as being

$$G = [I_k|P],$$
$$H = [-P^T|I_{(n-k)}].$$

For some particular P of size $k \times (n-k)$. It is also possible to define metrics over codes.

Definition 8 (Metric)**.** *A metric on a set* $\mathcal{X}$ *is a function (called distance function)*

$$d : \mathcal{X} \times \mathcal{X} \to [0, \infty),$$

where $\forall x, y, z \in \mathcal{X}$ *the following three axioms are satisfied:*

1. $d(x, y) = 0 \iff x = y$ *(identity of indiscernibles)*
2. $d(x, y) = d(y, x)$ *(symmetry)*
3. $d(x, y) \leq d(x, z) + d(z, y)$ *(triangle inequality)*

Definition 9 (Weight)**.** *Given a metric d on a set $\mathcal{X}$, the weight (function) $w : \mathcal{X} \to [0, \infty)$ induced by d is defined, $\forall x \in \mathcal{X}$, as $w(x) = d(x, 0) = d(0, x)$.*

One of the most useful metrics in Coding Theory is the Hamming Metric, defined bellow.

Definition 10 (Hamming Metric)**.** *The Hamming distance, d_h, of two vectors x and y from space $\mathbb{F}_2^n$ is the number of equivalent positions that differs between the two vectors, i.e.,*

$$d_h(x, y) = |\{i : 0 \leq i \leq n - 1, x_i \neq y_i\}|$$

Furthermore, the minimum distance and error correcting capability of a code are fundamental parameters related to codes, and thus are defined below.

Definition 11 (Minimum distance)**.** *Given a code $\mathcal{C}$ and an arbitrary metric d, e.g. the Hamming metric, the minimum distance of $\mathcal{C}$ with respect to d is*

$$\delta_d = \min_{c_1, c_2 \in \mathcal{C}} \{d(c_1, c_2)\} = \min_{c \in \mathcal{C}} \{w(c)\},$$

where $d(x, y)$ is the distance between x and y; and $w(x) = d(x, 0)$ is the weight function induced by d, for $x, y \in \mathbb{F}_q^n$.

Definition 12 (Error correcting capability)**.** *Given a code $\mathcal{C}$ with minimum distance δ_d with respect to a metric d, the code is t-error correcting if and only if $\delta_d \geq 2t + 1$.*

Since linear codes are defined over discrete vector spaces, some bounds regarding dimension, length and minimum distance must be observed. One of the most well-known bound in coding theory is defined in the sequence.

Theorem 1 (Singleton Bound)**.** *Given a $\mathcal{C}[n, k]$ code with minimum Hamming distance $\delta_{d_h} = d$, we have that*

$$d \leq n - k + 1$$

.

Proof. Let $\mathcal{C}[n,k]$ be a linear code with minimum Hamming distance $\delta_{d_h} = d$.

First observe that the number of codewords in $\mathcal{C}$ is q^k. This result is immediately, since we can construct all the codewords as linear combinations of vectors k (basis).

It is easy to see that all codewords $c \in \mathcal{C}$ are pairwise distinct. Thus, with we delete the first $d-1$ positions of each codeword, the resulting codewords must still be pairwise distinct since all of the original codewords in $\mathcal{C}$ have Hamming distance at least d from each other. Thus the cardinality of the altered code is the same as the original code.

The newly obtained code has length $n-(d-1) = n-d+1$ and, thus, there can be at most q^{n-d+1} of them. Since $\mathcal{C}$ was arbitrary, this bound must hold for the largest possible code with these parameters, thus

$$|\mathcal{C}| = q^k \leq q^{n-d+1} \rightarrow d \leq n-k+1$$

□

Definition 13 (Maximum Distance Separable (MDS) Code)**.** *A code $\mathcal{C}$ is said to be Maximum Distance Separable (MDS) if and only if its minimum distance achieves the extreme value of the Singleton bound, i.e.*

$$\delta_{d_h}(\mathcal{C}) = n-k+1$$

The two most important operations in algebraic coding theory are encoding and decoding. Those operations allow the correlation between an arbitrary space and the space defined by a code.

The encoding operating, given bellow, establishes the transformation of a given vector in space to a codeword.

Definition 14 (Encoding)**.** *Given a vector $v \in \mathbb{F}_q^k$, the encoding of v is the codeword c generated as $c = vG$.*

In turn, the decoding operation does the oppose transformation of the encoding, converting

a given codeword to a vector. There are many approaches to decode a codeword. The following presents the idea of the syndrome decoding approach.

Definition 15 (Syndrome)**.** *Given a code $\mathcal{C}$, the syndrome s of a vector $v \in \mathbb{F}_q^n$ is the value obtained from the operation $s = Hv^T$, where H is a parity check matrix for $\mathcal{C}$.*

Definition 16 (Decoding)**.** *Given a vector $v = c + e$ in $\mathbb{F}_q^n$, where c is a codeword in $\mathcal{C}$ and e is an error vector in $\mathbb{F}_q^n$, the decoding problem is to associate v with a correct codeword in $\mathcal{C}$.*

Note that $Hv^T = 0 \leftrightarrow e = 0$, this is, v is a codeword if and only if its syndrome is zero. The decoding process consists in eliminating the error vector e from v aiming to obtain c. There are several techniques for decoding the received vector v. One of these decoding techniques is the syndrome decoding, that considers that vectors affected by the same error belongs to the same coset.

Finally, some examples of code families with interesting extremal properties regarding minimum distance and error correction capability are given in the following.

Example 1 (Binary Hamming Code)**.** *The binary Hamming Code is a $\mathcal{C}[7, 4]$ code with length $n = 7$, dimension $k = 4$ and minimum distance $\delta_{d_h} = 3$. Its generator and parity matrices in systematic form are given bellow.*

$$G = \left[\begin{array}{cccc|ccc} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{array}\right],$$

and

$$H = \left[\begin{array}{cccc|ccc} 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 \end{array}\right].$$

Example 2 (Goppa Codes)**.** *Let t and n be two positive integers, with $t < n \leq q$. Let $g(x) \in \mathbb{F}_q[x]$ an irreducible polynomial of degree t in $\mathbb{F}_q$ and let $L = \{\alpha_1, \cdots, \alpha_n\}$ be an ordered set with n elements of $\mathbb{F}_q$ such that none of then is a root of g; that is, for all $i \in \{1, \ldots, n\}$, $g(\alpha_i) \neq 0$. The Goppa Code $\Gamma(g, L)$ is defined as*

$$\Gamma(g, L) = \left\{ c = (c_1, \cdots, c_n) \in \mathbb{F}_q^n, \textit{such that} \sum_{i=1}^{n} \frac{c_i}{x - L_i} \equiv 0 \mod g(x) \right\}.$$

The polynomial $g(x)$ is called Goppa's polynomial. The correcting capability of the code is t, the dimension is $n - mt$ and the minimum distance is at least $2t + 1$. Besides, one can find the Goppa's parity check matrix H using the matrices V and D, with $H = VD$,

$$V = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ \alpha_1 & \alpha_2 & \cdots & \alpha_n \\ \alpha_1^2 & \alpha_2^2 & \cdots & \alpha_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_1^{t-1} & \alpha_2^{t-1} & \cdots & \alpha_n^{t-1} \end{pmatrix}, \textit{ and } D = \begin{pmatrix} \frac{1}{g(\alpha_1)} & 0 & \cdots & 0 \\ 0 & \frac{1}{g(\alpha_2)} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{g(\alpha_n)} \end{pmatrix}.$$

4 MDS Matrices in Cryptography

As proposed by Shannon in "Communication Theory of Secrecy Systems" [Sha49], confusion and diffusion are key concepts in order to build secure criptosystems.

Confusion. Means that each binary digit (bit) of the ciphertext must depend on several parts of the key, obscuring the connections between the two.

Diffusion. Means that if we change a single bit of the plaintext, then (statistically) half of the bits in the ciphertext must change, and similarly, if we change one bit of the ciphertext, then approximately one half of the plaintext bits should change.

The main motivation for the usage of MDS codes in cryptography is that these structures achieve perfect diffusion. Indeed, as introduced by S. Vaudenay in [Vau95], MDS matrices

are isomorphic to multipermutations over an alphabet $\mathcal{Z}$, that, in it turns, achieve perfect diffusion.

Definition 17 (MDS matrix)**.** *A matrix M of order n is a MDS matrix if and only if every sub-matrix of M is non-singular (i.e., a superregular matrix)*

Equivalently, a matrix $M_{n\times n}$ is MDS if and only if

$$Y_{n\times 1} = M_{n\times n} X_{n\times 1} \implies \min_{X\neq 0}(W(Y) + W(X)) = n + 1$$

where $X = (x_0, x_1, \ldots, x_{n-1})^T$ and $Y = (y_0, y_1, \ldots, y_{n-1})^T$ are vectors in an arbitrary field and $W(X)$ is the number of non-zero elements of X.

Definition 18 (Multipermutation)**.** *An (r, n)-multipermutation over an alphabet $\mathcal{Z}$ is a function f from $\mathcal{Z}^r$ to $\mathcal{Z}^n$ such that two different $(r+n)$-tuples of the form $(x, f(x))$ cannot collide in any r positions.*

Example 3. *A $(1, n)$-multipermutation is a vector of n permutations over $\mathcal{Z}$.*

Example 4. *A $(2, 1)$-multipermutation is equivalent to a Latin square. Recall that a Latin square of order n with entries from an n-set X is an $n \times n$ array L in which every cell contains an element of X such that every row of L is a permutation of X and every column of L is a permutation of X (see [Sti04] for more details).*

An equivalent definition of multipermutation is that the set of all $(r + n)$-tuples of the form $(x, f(x))$ is an linear error correcting code with minimum distance $n + 1$, which is the maximal possible (recall the Singleton bound, Theorem 1. For more details, see [HP03, Section 2.4]). In the case of a linear function f, this is exactly the definition of MDS codes with parameters $[n + r, r, n + 1]$.

More generally, a (r, n)-multipermutation is equivalent to a $(|\mathcal{Z}|^r, r+n, |\mathcal{Z}|, r)$-orthogonal array. Recall that an orthogonal array $\mathrm{OA}(k, n)$, with $k \geq 2$ and $n \geq 1$, is an $n^2 \times k$ array, A, with entries from a set X of cardinality n such that, within any two columns of A, every

ordered pair of symbols from X occurs in exactly one row of A. In this case $n = |\mathcal{Z}|$ and $k = r$.

A multipermutation also corresponds to the notion of *perfect local randomizer* introduced by U. Maurer and J. Massey in [MM00].

Definition 19 (Sequence generator)**.** *A (k, n) sequence generator G is a function $G : \{0,1\}^k \to \{0,1\}^n$.*

Definition 20 (Perfect local randomizer)**.** *A perfect local randomizer is a sequence generator that stretches a (binary) random sequence of length k to a pseudo random sequence of length n such that every subset of e or less bits of the generated sequence is a set of independent random bits.*

The concept of a perfect local randomizer also corresponds to what is known in combinatorics as an orthogonal array [Sti04].

A multipermutation performs a perfect diffusion in the sense that changing t of the inputs changes at least $n - t + 1$ of the outputs. In fact, it corresponds to the notion of perfect local randomizer [MM00] with optimal parameter for collisions. Thus, if a function is not a multipermutation, one can find several values such that both few inputs and few outputs are changed.

The design of multipermutations over a large alphabet is a very difficult problem, as the design of mutually orthogonal latin squares (MOLS) is a well-known difficult one [Sti04]. The only powerful method seems to use an MDS code combined with several permutations at each coordinate.

4.1 SPN and P-SPN ciphers

In cryptography, a Substitution-Permutation Network (SPN) is a series of linked mathematical operations used in many block cipher algorithms such as AES (Rijndael)[DR02]. It is essentially composed by a substitution layer (S-boxes), that substitutes a small block of

bits (the input of the S-box) with another block of bits (the output of the S-box) by means of a non-linear transformation; and a permutation layer (P-boxes), that takes the outputs of all the S-boxes of one round, permutes the bits, and feeds them into the S-boxes of the next round, as discussed in Section 2.3.1. This can be summarized in Figure 3.

Let $\mathbb{F}$ be a finite field of size q^n with q a prime number. Formally, SPN ciphers can be defined as follows.

Definition 21 (SPN Ciphers)**.** *An r-rounds SPN cipher is an application $E_k^r : \mathbb{F}^t \to \mathbb{F}^t$, where $k \in \mathbb{F}^t$, so that for every message $x = (x_1, x_2, \ldots, x_t) \in \mathbb{F}^t$, the encryption is defined by*

$$E_k^r(x) = (F_r \circ \ldots \circ F_0)(x + k^{(0)}),$$

where $F_i : \mathbb{F}^t \to \mathbb{F}^t$ is defined as $F_i(x) = R(x) + k^{(i)}$, for $i \in [1, r]$ and $k^{(0)}, \ldots, k^{(t)} \in \mathbb{F}^t$ are the round keys.

In the case of an SPN permutation, the secret round keys are just replaced by public round constants. We denote by R the composition of the S-box and the linear layer, i.e., $R : \mathbb{F}^t \to \mathbb{F}^t$ with

$$R(X) = (M \circ S)(x) = M(S_1(x_1), \ldots, S_t(x_t)),$$

where $S_i : \mathbb{F}^t \to \mathbb{F}^t$, for $i \in [1, t]$ is a nonlinear polynomial S-box and $M : \mathbb{F}^t \to \mathbb{F}^t$ denotes an invertible non-trivial linear layer defined by the multiplication with a matrix.

Definition 22 (Non-trivial linear layer)**.** *A linear layer $M : \mathbb{F} \to \mathbb{F}$ is non-trivial if it ensures full diffusion (in the sense that each bit of the output depends on each bit of the input and vice versa) after a finite number of rounds.*

It follows immediately from Definition 22 that MDS matrices composes a class of non-trivial linear layers for SPN schemes, since they provide full diffusion [Vau95].

The SPN application on a cryptoscheme can be computational expansive. Aiming to reduce that cost, it was introduced the notion of partial substitution-permutation network

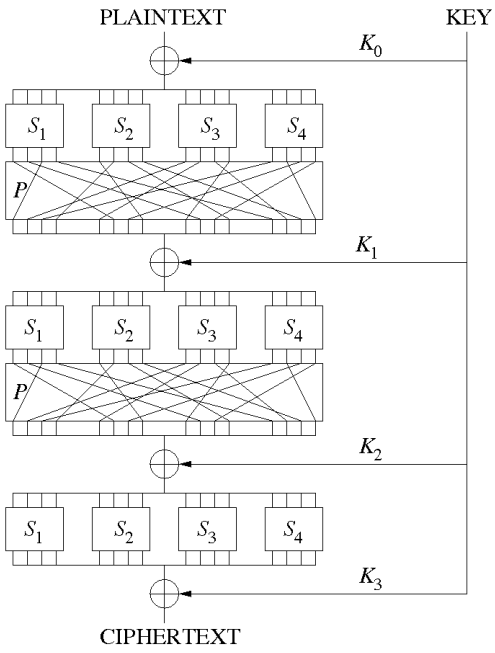

Figure 3: 3-rounds SPN.

(P-SPN), where some SPN's S-boxes are chosen to be identity functions, which reduces the cost of performing non-linear transformations for all S-boxes.

Definition 23 (P-SPN Ciphers)**.** *A r-rounds P-SPN cipher is an application $E_k^r : \mathbb{F}^t \to \mathbb{F}^t$ on a message x in $\mathbb{F}^t$, where $k \in \mathbb{F}^t$, so that for every input $x = (x_1, x_2, \ldots, x_t)$ the encryption is defined by $E_k^r(x) = (F_r \circ \ldots \circ F_0)(x + k^{(0)})$, where $F_i : \mathbb{F}^t \to \mathbb{F}^t$ is defined as $F_i(x) = R(x) + k^{(i)}$, for $i \in [1, r]$ and the round keys $k^{(0)}, \ldots, k^{(t)} \in \mathbb{F}^t$. In the case of an SPN permutation, the secret round keys are just replaced by public round constants. We denote by R the composition of the S-box and the linear layer, i.e., $R : \mathbb{F}^t \to \mathbb{F}^t$ with*

$$R(X) = (M \circ S)(x) = M(S_1(x_1), \ldots, S_s(x_s), I_{s+1}, \ldots, I_t),$$

where $s \in [1, t)$ is the number of active S-boxes, $S_i : \mathbb{F} \to \mathbb{F}$, for $i \in [1, t]$ is a nonlinear polynomial S-box; $I_{s+1} = \cdots = I_t$ are identity functions; and $M : \mathbb{F} \to \mathbb{F}$ denotes an invertible non-trivial linear layer defined by the multiplication with a matrix.

The main consequence of simply applying P-SPNs instead of SPNs is security. Aiming to reduce the impacts on security, it was proposed in [GLR+19] the so called HADES strategy, that combines the usage of both SPN and P-SPN strategies, as can be seen in Figure 4.

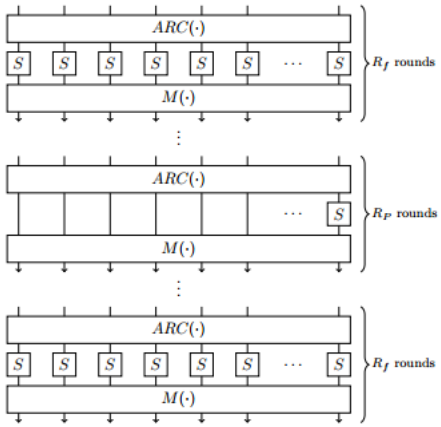

Figure 4: HADES strategy.

Moderns cryptosystems like Poseidon [GKR+19], LowMC [ARS+16] and Zorro [GGNPS13] use the HADES strategy, combined with some exhaustive repetition strategy (e.g. the calculation of several rounds on a Merkle tree; see [Mer90] for more details) to ensure security. However, the performance of those schemes can be seriously affected. This motivates the study of linear layers on P-SPN schemes, aiming to improve security and performance simultaneously.

4.2 Invariant Subspaces and Subspace Trails

Before studying weak linear layers, let us present the definitions of invariant subspace (introduced in [LAAZ11]) and subspace trails (introduced in [GRR16]), that lead to several attacks on SPN cryptosystems.

Definition 24 (Invariant Subspace Trails)**.** *Let K_{weak} be a set of keys and $k \in K_{weak}$, with $k = (k^{(0)}, k^{(1)}, \ldots, k^{(r)})$, where each $k^{(j)}$, $j \in [0, r]$ is the j-th round key. For $k \in K_{weak}$, the subspace $\mathcal{IS}$ generates an invariant subspace trail of length r for the round function $R_k(\cdot) =$*

$R(\cdot)+k$ if for each $i \in [1,r]$ there exists a non-empty set $A_i \subseteq \overline{\mathcal{IS}}$ (the complementary subspace of $\mathcal{IS}$) for which

$$\forall a_i \in A_i, \exists a_{i+1} \in A_{i+1} : R_{k^{(i)}}(\mathcal{IS}+a_i) = R(\mathcal{IS}+a_i) = \mathcal{IS}+a_{i+1}.$$

All keys in the set K_{weak} are weak keys and can be exploited by a non-authorized party.

Definition 25 (Subspace Trail)**.** *Let $(\mathcal{U}_1,\ldots,\mathcal{U}_{r+1})$ denote a set of $r+1$ subspaces with $dim(\mathcal{U}_i) \leq dim(\mathcal{U}_{i+1})$. If for each $i \in [1,r]$ we have*

$$\forall a_i \in \overline{\mathcal{U}_i}, \exists a_{i+1} \in \overline{\mathcal{U}_{i+1}} : R^{(i)}(\mathcal{U}_i + a_i) \subseteq \mathcal{U}_{i+1} + a_{i+1},$$

then $(\mathcal{U}_1,\ldots,\mathcal{U}_{r+1})$ is a subspace trail of length r for the function $F(\cdot) = (R^{(r)} \circ \cdots \circ R^{(1)})(\cdot)$. If all the previous relations hold with equality, the trail is a constant-dimensional subspace trail.

Definition 26 (Iterative (Constant-Dimensional) Subspace Trails)**.** *Let $\{\mathcal{V}_1,\mathcal{V}_2,\ldots,\mathcal{V}_r\}$ be a constant-dimensional subspace trail for r rounds. This subspace trail is an infinitely long iterative (constant-dimensional) subspace trail of period r for the considered scheme function*

$$F(\cdot) = (R^{(r)} \circ \cdots \circ R^{(1)})(\cdot)$$

if it repeats itself an infinite number of times, i.e., if

$$\{\mathcal{V}_1,\mathcal{V}_2,\ldots,\mathcal{V}_r,\mathcal{V}_1,\mathcal{V}_2,\ldots,\mathcal{V}_r,\mathcal{V}_1,\mathcal{V}_2,\ldots,\mathcal{V}_r,\ldots\}$$

is an infinitely long subspace trail.

Clearly, an invariant subspace trail is also an iterative subspace trail for P-SPN schemes under the assumptions that $A_i = \mathbb{F}^t$ and that the set K_{weak} is equal to the set of all possible keys; while not every iterative subspace trail is also an invariant subspace trail. At the same time, the following result holds.

Proposition 1. *Working over* $\mathbb{F}^t$ *, let* $\{\mathcal{V}_1, \mathcal{V}_2, \ldots, \mathcal{V}_r\}$ *be an infinitely long iterative subspace trail of period* r*. If* $dim(Span\{\mathcal{V}_1, \mathcal{V}_2, \ldots, \mathcal{V}_r\}) < t$*, then* $Span\{\mathcal{V}_1, \mathcal{V}_2, \ldots, \mathcal{V}_r\}$ *generates an infinitely long invariant subspace trail.*

Proof. The subspace $\text{Span}\{\mathcal{V}_1, \mathcal{V}_2, \ldots, \mathcal{V}_r\}$ is invariant since each coset of $\mathcal{V}_i$ is mapped into a coset of $\mathcal{V}_{i+1}$, by definition. □

Before continuing, we briefly mention the link between truncated differential trails and subspace trails. Differential attacks [BS91] exploit the fact that pairs of plaintexts with certain differences yield other differences in the corresponding ciphertexts with a non-uniform probability distribution. A variant of this attack/distinguisher is the truncated differential one [Knu95], in which the attacker can predict only part of the difference between pairs of texts. Using the subspace terminology, given pairs of plaintexts that belong to the same coset of a subspace $\mathcal{X}$, one considers the probability that the corresponding ciphertexts belong to the same coset of a subspace $\mathcal{Y}$ to set up an attack. In particular, note that two texts are in the same coset of a given subspace if and only if their difference belongs to such a subspace:

$$x, y \in \mathcal{V} + \alpha \text{ if and only if } x - y \in \mathcal{V}.$$

The relation between truncated differential trails and subspace trails has been studied in details in [LTW18] and [BLN16]. Finally, impossible differential and truncated impossible differential attacks based on differentials that hold with probability zero have been studied in [BBS99].

4.3 Subspace Trails for P-SPN Schemes (Inactive S-boxes): A necessary and sufficient condition

Recalling the fact that the nonlinear layer is only partial in P-SPN schemes, with $s \geq 1$ (active) S-boxes and $t - s \geq 1$ identity functions (i.e., inactive S-boxes), it is always possible to find an initial subspace such that no S-box is active (at least) in the first $\left\lfloor \frac{t-s}{s} \right\rfloor$ rounds.

Indeed, assuming that s S-boxes are applied to the first s blocks of a text and choosing texts in a same coset of $\mathcal{S} = \text{Span}\{v_1, \ldots, v_d\}$ s.t. $d = \dim(\mathcal{S}) \geq t - s\left\lfloor \frac{t-s}{s} \right\rfloor$ and $1 \leq i \leq \lfloor \frac{t-s}{s} \rfloor$, $1 \leq j \leq d : (M^{i-1}v_j)_{[1,2,\ldots,s]} = (0, 0, \ldots, 0) \in \mathbb{F}^s$; it follows that no S-box is active in the first $\left\lfloor \frac{t-s}{s} \right\rfloor$ rounds. This result is formalized in the following definition.

Definition 27. *Consider a P-SPN scheme over $\mathbb{F}^t$ as in Definition 23. Then, we define the subspace $\mathcal{S}^{(.)}$ as $\mathcal{S}^{(0)} = \mathbb{F}^t$, and for $i \geq 1$*

$$\mathcal{S}^{(i)} := \{v \in \mathbb{F}^t : (M^j \cdot v)_{[1,2,\ldots,s]} = (0, 0, \ldots, 0) \in \mathbb{F}^s, j < i\}.$$

Theorem 2.
$\mathcal{S}^{(i+1)} := \{v \in \mathcal{S}^{(i)} : (M \cdot v)_{[1,2,\ldots,s]} = (0, 0, \ldots, 0) \in \mathbb{F}^s, j < i\} = \mathcal{S}^{(i)} \cap (M^{-(i-1)}\mathcal{S}^{(1)}) \subseteq \mathcal{S}^{(i)}$.

Proof. Given $\mathcal{S}^{(1)} = \text{Span}\{e_{s+1}, \ldots, e_t\}$, we observe that $(M \cdot x)_{[1,\ldots,s]} = (0, 0, \ldots, 0) \in \mathbb{F}^s$ if and only if $(M \cdot x) \in \mathcal{S}^{(1)}$, i.e., $x \in M^{-1}\mathcal{S}^{(1)}$. Thus

$$\mathcal{S}^{(i+1)} = \mathcal{S}^{(1)} \cap M^{-1}\mathcal{S}^{(1)} \cap \cdots \cap M^{-i}\mathcal{S}^{(1)}.$$

Hence, given $x \in \mathbb{F}^t$, it follows that $x \in \mathcal{S}^{(i+1)}$ if and only if $x \in \mathcal{S}^{(i)}$ and $x \in (M^{-(i-1)}\mathcal{S}^{(1)})$.

□

Theorem 3 ([GRS20])**.** *Given a P-SPN scheme with s S-boxes defined as Definition 23, let $M \in \mathbb{F}^{t\times t}$ be an invertible matrix. A subspace $\mathcal{IS}$, where $1 \leq dim(\mathcal{IS}) < t$, generates an infinitely long invariant subspace trail (with no active S-boxes) if and only if there exists $i \geq 1$ s.t. $\mathcal{S}^{(i)} = M.\mathcal{S}^{(i)}$ and $\mathcal{IS} \subseteq \mathcal{S}^{(i)}$. Similarly, a subspace $\mathcal{IS}$, where $1 \leq dim(\mathcal{IS}) < t$, generates an infinitely long iterative (non-invariant) subspace trail of period $l \geq 2$ (with no active S-boxes) if and only if there exists $i \geq l$ s.t. $\mathcal{S}^{(i)} = M^l.\mathcal{S}^{(i)}$ and $\mathcal{IS} \subseteq \mathcal{S}^{(i)}$.*

4.3.1 Non-existence of infinitely long invariant subspace trails for MDS linear layers

Based on the result presented by Theorem 3, in this section we will proof that a necessary (but not sufficient) condition for the non-existence of infinitely long invariant subspace trails is that the linear layer must be MDS. This proof reinforces the good fit of MDS matrices for certain cryptoschemas, since, in addition to providing perfect diffusion, they provide resistance against invariant subspace attacks.

Definition 28 (Generalized Inverse)**.** *Let $A \in \mathbb{F}^{m\times n}$ be an arbitrary matrix. The matrix $\bar{A} \in \mathbb{F}^{n\times m}$ is a generalized inverse of A if it satisfies the equation $A\bar{A}A = A$.*

Lemma 1. *The general solution of any homogeneous system $AY = 0$, with $A \in \mathbb{F}^{m\times n}$ and $Y \in \mathbb{F}^{n\times r}$, can be written as $Y = (I_n - \bar{A}A)U$, where $U \in \mathbb{F}^{n\times r}$ is arbitrary.*

Proof. Considering $Y = (I_n - \bar{A}A)U$, we have that $A \cdot Y = A(I_n - \bar{A}A)U = (A - A\bar{A}A)U = (A - A)U = O \cdot U = O$ for any matrix $U \in \mathbb{F}^{n\times r}$. □

Lemma 2. *[MS74] Given the matrices A, B, C with the appropriate dimensions, then*

$$\text{rank}([A, B]) = \text{rank}(A) + \text{rank}(B - A\bar{A}B), \tag{1}$$

and

$$\text{rank}\left(\begin{bmatrix} A \\ C \end{bmatrix}\right) = \text{rank}(A) + \text{rank}(C - C\bar{A}A). \tag{2}$$

With the Definition 28 and Lemmas 1 and 2, we can prove the following result.

Lemma 3. *Let $A_i \in \mathbb{F}^{m\times n}, i \in [1, k]$ be a set of k matrices. The dimension of the*

intersection of the column subspaces $\mathrm{col}(A_i)$ *is given by*

$$\dim\left(\bigcap_{i=1}^{k}\mathrm{col}(A_i)\right) = \sum_{i=1}^{k}\mathrm{rank}(A_i) - \mathrm{rank}\begin{pmatrix} A_1 & A_2 & O & \dots & O \\ A_1 & O & A_3 & \dots & O \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ A_1 & O & O & \dots & A_k \end{pmatrix}.$$

Proof. Let $X \in \mathbb{F}^{m\times 1}$ be an arbitrary element of $\bigcap_{i=1}^{k}\mathrm{col}(A_i)$. We have that there exist $X_i \in \mathbb{F}^{n\times 1}$, for $1 \leq i \leq k$, such that

$$X = A_1X_1 = A_2X_2 = \cdots = A_kX_k, \tag{3}$$

this is, $X \in \bigcap_{i=1}^{k}\mathrm{col}(A_i)$ if and only if $X \in \mathrm{col}(A_i)$ for all $i = 1, \dots, k$. The expression in Equation (3) can be seen as a matrix system of the form:

$$\begin{pmatrix} I_m & -A_1 & O & \dots & O \\ I_m & O & -A_2 & \dots & O \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ I_m & O & O & \dots & -A_k \end{pmatrix}\begin{pmatrix} X \\ X_1 \\ \vdots \\ X_k \end{pmatrix} = 0 \tag{4}$$

Equivalently, the system in (4) can be denoted as $MY = 0$, where $M \in \mathbb{F}^{km\times t}$ and $Y \in \mathbb{F}^{t\times 1}$ for $t = (k+1)n$. Thus, according to Lemma 1, the general solution for the system $MY = 0$ presented on Equation (4) is $Y = (I_t - \bar{M}M)U$, where $U \in \mathbb{F}^{t\times 1}$ is arbitrary. Therefore, the general expression for X is

$$X = (I_m, O, \dots, O)Y = (I_m, O, \dots, O)(I_t - \bar{M}M)U.$$

Since $X \in \bigcap_{i=1}^{k}\mathrm{col}(A_i)$ is arbitrary, the dimension of $\bigcap_{i=1}^{k}\mathrm{col}(A_i)$ can be calculated using

Lemma 2 as

$$\dim\left(\bigcap_{i=1}^{k}\mathrm{col}(A_i)\right) = \mathrm{rank}((I_m,O,\ldots,O)(I_t-\bar{M}M)) = \mathrm{rank}((I_m,O,\ldots,O)-(I_m,O,\ldots,O)\bar{M}M))$$

$$= \mathrm{rank}\left(\begin{bmatrix} I_m & O & \ldots & O \\ & & M & \end{bmatrix}\right) - \mathrm{rank}(M)$$

$$= \mathrm{rank}\begin{pmatrix} I_m & O & \ldots & O \\ I_m & -A_1 & \ldots & O \\ \vdots & \vdots & \ddots & \vdots \\ I_m & O & \ldots & -A_k \end{pmatrix} - \mathrm{rank}\begin{pmatrix} I_m & -A_1 & \ldots & O \\ \vdots & \vdots & \ddots & \vdots \\ I_m & O & \ldots & -A_k \end{pmatrix}$$

$$= m + \mathrm{rank}\begin{pmatrix} -A_1 & \ldots & O \\ \vdots & \ddots & \vdots \\ O & \ldots & -A_k \end{pmatrix} - m - \mathrm{rank}\begin{pmatrix} A_1 & -A_2 & O & \ldots & O \\ A_1 & O & -A_3 & \ldots & O \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ A_1 & O & O & \ldots & -A_k \end{pmatrix}$$

$$= \sum_{i=1}^{k}\mathrm{rank}(A_i) - \mathrm{rank}\begin{pmatrix} A_1 & A_2 & O & \ldots & O \\ A_1 & O & A_3 & \ldots & O \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ A_1 & O & O & \ldots & A_k \end{pmatrix}.$$

□

Now, with Theorem 3 in hands, we want to characterize a linear layer M in order do avoid the generation of an infinitely long invariant subspace trail (with no active S-boxes). That is, we want to find a condition that ensures for $i \geq 1, \mathcal{S}^{(i)} \neq M.\mathcal{S}^{(i)}$. Hence, it holds for $i \geq 1, \dim(\mathcal{S}^{(i)}) \neq \dim(M \cdot \mathcal{S}^{(i)})$ implying $\mathcal{S}^{(i)} \neq M \cdot \mathcal{S}^{(i)}$

Theorem 4. *If the condition of Theorem 3 won't hold, i.e., $\nexists i \geq 1$ s.t. $\mathcal{S}^{(i)} = M.\mathcal{S}^{(i)}$ and $\mathcal{IS} \subseteq \mathcal{S}^{(i)}$; then M is MDS.*

Proof. From Lemma 2, we have that for $i \geq 1$,

$$\mathcal{S}^{(i)} = \mathcal{S}^{(1)} \cap M^{-1}\mathcal{S}^{(1)} \cap (M^{-1})^2\mathcal{S}^{(1)} \cap \cdots \cap (M^{-1})^{(i-2)}\mathcal{S}^{(1)} \cap (M^{-1})^{(i-1)}\mathcal{S}^{(1)}, \tag{5}$$

and

$$M \cdot \mathcal{S}^{(i)} = M \cdot \mathcal{S}^{(1)} \cap \mathcal{S}^{(1)} \cap (M^{-1})\mathcal{S}^{(1)} \cap \cdots \cap (M^{-1})^{(i-3)}\mathcal{S}^{(1)} \cap (M^{-1})^{(i-2)}\mathcal{S}^{(1)}. \tag{6}$$

Follows immediately from Lemma 3 and Equations (5) and (6) that

$$\begin{aligned}\dim(\mathcal{S}^{(i)}) =& \dim\left(\mathcal{S}^{(1)} \cap M^{-1}\mathcal{S}^{(1)} \cap (M^{-1})^2\mathcal{S}^{(1)} \cap \cdots \cap (M^{-1})^{(i-2)}\mathcal{S}^{(1)} \cap (M^{-1})^{(i-1)}\mathcal{S}^{(1)}\right) \\ =& \operatorname{rank}(\mathcal{S}^{(1)}) + \operatorname{rank}(M^{-1}\mathcal{S}^{(1)}) + \operatorname{rank}((M^{-1})^2\mathcal{S}^{(1)}) + \\ &+ \cdots + \operatorname{rank}((M^{-1})^{(i-2)}\mathcal{S}^{(1)}) + \operatorname{rank}((M^{-1})^{(i-1)}\mathcal{S}^{(1)}) - \\ &- \operatorname{rank}\begin{pmatrix} \mathcal{S}^{(1)} & M^{-1}\mathcal{S}^{(1)} & 0 & 0 & \dots & 0 \\ \mathcal{S}^{(1)} & 0 & (M^{-1})^2\mathcal{S}^{(1)} & 0 & \dots & 0 \\ \vdots & 0 & 0 & 0 & \ddots & 0 \\ \mathcal{S}^{(1)} & 0 & 0 & 0 & \dots & (M^{-1})^{(i-1)}\mathcal{S}^{(1)} \end{pmatrix},\end{aligned} \tag{7}$$

and that

$$\dim(M\mathcal{S}^{(i)}) = \dim\left(M\mathcal{S}^{(1)} \cap \mathcal{S}^{(1)} \cap (M^{-1})\mathcal{S}^{(1)} \cap \cdots \cap (M^{-1})^{(i-3)}\mathcal{S}^{(1)} \cap (M^{-1})^{(i-2)}\mathcal{S}^{(1)}\right)$$

$$= \text{rank}(\mathcal{S}^{(1)}) + \text{rank}(M\mathcal{S}^{(1)}) + \text{rank}(M^{-1}\mathcal{S}^{(1)}) + \text{rank}((M^{-1})^2\mathcal{S}^{(1)}) + \ldots + \tag{8}$$

$$+ \text{rank}((M^{-1})^{(i-2)}\mathcal{S}^{(1)}) - \text{rank}\begin{pmatrix} \mathcal{S}^{(1)} & M\mathcal{S}^{(1)} & 0 & 0 & \ldots & 0 \\ \mathcal{S}^{(1)} & 0 & (M^{-1})\mathcal{S}^{(1)} & 0 & \ldots & 0 \\ \vdots & 0 & 0 & 0 & \ddots & 0 \\ \mathcal{S}^{(1)} & 0 & 0 & 0 & \ldots & (M^{-1})^{(i-2)}\mathcal{S}^{(1)} \end{pmatrix}.$$

Hence, according to Equations (7) and (8), in order to have $\dim(\mathcal{S}^{(i)}) \neq \dim(M.\mathcal{S}^{(i)})$ implying $\mathcal{S}^{(i)} \neq M \cdot \mathcal{S}^{(i)}$, for $i \geq 1$, we need

$$\text{rank}((M^{-1})^{(i-1)}\mathcal{S}^{(1)}) - \text{rank}\begin{pmatrix} \mathcal{S}^{(1)} & M^{-1}\mathcal{S}^{(1)} & 0 & 0 & \ldots & 0 \\ \mathcal{S}^{(1)} & 0 & (M^{-1})^2\mathcal{S}^{(1)} & 0 & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathcal{S}^{(1)} & 0 & 0 & 0 & \ldots & (M^{-1})^{(i-1)}\mathcal{S}^{(1)} \end{pmatrix} \neq$$

$$\neq \text{rank}(M\mathcal{S}^{(1)}) - \text{rank}\begin{pmatrix} \mathcal{S}^{(1)} & M\mathcal{S}^{(1)} & 0 & 0 & \ldots & 0 \\ \mathcal{S}^{(1)} & 0 & (M^{-1})\mathcal{S}^{(1)} & 0 & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathcal{S}^{(1)} & 0 & 0 & 0 & \ldots & (M^{-1})^{(i-2)}\mathcal{S}^{(1)} \end{pmatrix}. \tag{9}$$

Applying Lemma 2, we can simplify Equation (9) as follows:

$$\text{rank}((M^{-1})^{(i-1)}\mathcal{S}^{(1)}) - \text{rank}\begin{pmatrix} \mathcal{S}^{(1)} & M^{-1}\mathcal{S}^{(1)} & 0 & 0 & \dots & 0 \\ \mathcal{S}^{(1)} & 0 & (M^{-1})^2\mathcal{S}^{(1)} & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathcal{S}^{(1)} & 0 & 0 & 0 & \dots & (M^{-1})^{(i-1)}\mathcal{S}^{(1)} \end{pmatrix}$$

$$= \text{rank}((M^{-1})^{(i-1)}\mathcal{S}^{(1)}) - \text{rank}(\mathcal{S}^{(1)}) - \text{rank}\begin{pmatrix} M^{-1}\mathcal{S}^{(1)} & (M^{-1})^2\mathcal{S}^{(1)} & \dots & 0 & 0 \\ M^{-1}\mathcal{S}^{(1)} & 0 & (M^{-1})^3\mathcal{S}^{(1)} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ M^{-1}\mathcal{S}^{(1)} & 0 & 0 & \dots & (M^{-1})^{(i-1)}\mathcal{S}^{(1)} \end{pmatrix}$$

$$\neq \text{rank}(M\mathcal{S}^{(1)}) - \text{rank}\begin{pmatrix} \mathcal{S}^{(1)} & M\mathcal{S}^{(1)} & 0 & 0 & \dots & 0 \\ \mathcal{S}^{(1)} & 0 & (M^{-1})\mathcal{S}^{(1)} & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathcal{S}^{(1)} & 0 & 0 & 0 & \dots & (M^{-1})^{(i-2)}\mathcal{S}^{(1)} \end{pmatrix}$$

$$= \text{rank}(M\mathcal{S}^{(1)}) - \text{rank}(\mathcal{S}^{(1)}) - \text{rank}\begin{pmatrix} M\cdot\mathcal{S}^{(1)} & (M^{-1})\mathcal{S}^{(1)} & \dots & 0 & 0 \\ M\cdot\mathcal{S}^{(1)} & 0 & (M^{-1})^2\mathcal{S}^{(1)} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ M\cdot\mathcal{S}^{(1)} & 0 & 0 & \dots & (M^{-1})^{(i-2)}\mathcal{S}^{(1)} \end{pmatrix}$$

$$= \text{rank}(M\mathcal{S}^{(1)}) - \text{rank}(\mathcal{S}^{(1)}) - rank(M\mathcal{S}^{(1)})$$
$$- \text{rank}\begin{pmatrix} M^{-1}\mathcal{S}^{(1)} & (M^{-1})^2\mathcal{S}^{(1)} & \dots & 0 & 0 \\ M^{-1}\mathcal{S}^{(1)} & 0 & (M^{-1})^3\mathcal{S}^{(1)} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ M^{-1}\mathcal{S}^{(1)} & 0 & 0 & \dots & (M^{-1})^{(i-2)}\mathcal{S}^{(1)} \end{pmatrix}$$

Now, let us denote:

$$A = \begin{pmatrix} M^{-1}\mathcal{S}^{(1)} & (M^{-1})^2\mathcal{S}^{(1)} & 0 & \dots & 0 \\ M^{-1}\mathcal{S}^{(1)} & 0 & (M^{-1})^3\mathcal{S}^{(1)} & \dots & 0 \\ \vdots & & & \ddots & 0 \\ M^{-1}\mathcal{S}^{(1)} & 0 & 0 & (M^{-1})^{(i-2)}\mathcal{S}^{(1)} & 0 \end{pmatrix}$$

and

$$C = \begin{pmatrix} M^{-1}\mathcal{S}^{(1)} & 0 & 0 & \dots & (M^{-1})^{(i-1)}\mathcal{S}^{(1)} \end{pmatrix}.$$

A generalized inverse matrix for A is given by

$$\bar{A} = \begin{pmatrix} 0 & \dots & 0 \\ \overline{(M^{-1})^2\mathcal{S}^{(1)}} & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & 0 & \overline{(M^{-1})^{(i-2)}\mathcal{S}^{(1)}} \end{pmatrix}$$

so that

$$\bar{A}A = \begin{pmatrix} I & 0 & \dots & 0 \\ 0 & I & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & I \end{pmatrix}$$

. Thus, by Lemma 2, we have that

$$\text{rank}\left(\begin{bmatrix} A \\ C \end{bmatrix}\right) = \text{rank}(A) + \text{rank}(C - C\bar{A}A)$$

$$= \text{rank}\left(\left[\begin{array}{ccccc} M^{-1}\mathcal{S}^{(1)} & (M^{-1})^2\mathcal{S}^{(1)} & 0 & \dots & 0 \\ M^{-1}\mathcal{S}^{(1)} & 0 & (M^{-1})^3\mathcal{S}^{(1)} & \dots & 0 \\ \vdots & & & \ddots & 0 \\ M^{-1}\mathcal{S}^{(1)} & 0 & 0 & (M^{-1})^{(i-2)}\mathcal{S}^{(1)} & 0 \\ \hline M^{-1}\mathcal{S}^{(1)} & 0 & 0 & \dots & (M^{-1})^{(i-1)}\mathcal{S}^{(1)} \end{array}\right]\right)$$

$$= \text{rank}\begin{pmatrix} M^{-1}\mathcal{S}^{(1)} & (M^{-1})^2\mathcal{S}^{(1)} & 0 & \dots & 0 \\ M^{-1}\mathcal{S}^{(1)} & 0 & (M^{-1})^3\mathcal{S}^{(1)} & \dots & 0 \\ \vdots & & & \ddots & 0 \\ M^{-1}\mathcal{S}^{(1)} & 0 & 0 & (M^{-1})^{(i-2)}\mathcal{S}^{(1)} & 0 \end{pmatrix}$$

$$+ \text{rank}\left(\begin{array}{c} (M^{-1}\mathcal{S}^{(1)} \quad 0 \quad 0 \quad \dots \quad (M^{-1})^{(i-1)}) - (M^{-1}\mathcal{S}^{(1)} \quad 0 \quad 0 \quad \dots \quad (M^{-1})^{(i-1)}) \begin{pmatrix} I & 0 & \dots & 0 \\ 0 & I & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & I \end{pmatrix} \\ \\ \end{array}\right)$$

$$= \text{rank}\begin{pmatrix} M^{-1}\mathcal{S}^{(1)} & (M^{-1})^2\mathcal{S}^{(1)} & \dots & 0 \\ \vdots & & \ddots & 0 \\ M^{-1}\mathcal{S}^{(1)} & 0 & \dots & (M^{-1})^{(i-2)}\mathcal{S}^{(1)} \end{pmatrix}$$

$$+ \text{rank}\left((M^{-1}\mathcal{S}^{(1)} \quad 0 \quad 0 \quad \dots \quad (M^{-1})^{(i-1)}) - (M^{-1}\mathcal{S}^{(1)} \quad 0 \quad 0 \quad \dots \quad (M^{-1})^{(i-1)})\right)$$

$$= \text{rank}\begin{pmatrix} M^{-1}\mathcal{S}^{(1)} & (M^{-1})^2\mathcal{S}^{(1)} & \dots & 0 \\ \vdots & & \ddots & 0 \\ M^{-1}\mathcal{S}^{(1)} & 0 & \dots & (M^{-1})^{(i-2)}\mathcal{S}^{(1)} \end{pmatrix}$$

Since all the terms $\text{rank}(\mathcal{S}^{(1)}), \ldots, \text{rank}(M^{-1})^{(i-2)}\mathcal{S}^{(1)})$ cancel each other for both Equations (7) and (8), it follows that Equation (9) can be rewritten as

$$\text{rank}\left((M^{-1})^{(i-1)}\mathcal{S}^{(1)}\right) \neq 0, \forall i > 1 \tag{10}$$

Thus, recalling Definition 17, Equation (10) states that if M is MDS then the condition of Theorem 3 won't hold. □

5 Construction of MDS Matrices

Besides Theorem 4 ensuring that a necessary condition for the non-existence for infinitely long iterative (non-invariant) subspaces trails is the usage of a MDS linear layer, this condition is not sufficient. This is, even using MDS matrices as linear layer it is still possible (although very unlikely) to find an invariant subspace based attack.

Thus, the construction of MDS matrices that are sufficient itself to avoid infinitely long iterative subspaces trails remains an open problem [GRS20]. Even so, this section presents some MDS construction techniques proposed [CCR13] that can be investigated and/or adapted to fill this gap.

Definition 29 (Superregular b-block matrix)**.** *A matrix $A = [A_{ij}] \in Mat_{bm\times bt}(\mathbb{F}_q)$, where each A_{ij} has size $b \times b$ is a superregular b-block matrix if every submatrix of A consisting of full blocks matrices A_{ij} is non-singular over $\mathbb{F}_q$.*

Theorem 5 ([CCR13])**.** *Let $G = \begin{bmatrix} I_k & A \end{bmatrix}$ be a $kb \times nb$ systematic generator matrix of an $\mathbb{F}_q$-linear code $\mathcal{C}_{\mathbb{F}_q^b}$ with parameters $[n, k]$. Then $\mathcal{C}_{\mathbb{F}_q^b}$ is MDS if and only if A is a superregular b-block matrix.*

Definition 30 (Companion Matrix)**.** *Given a polynomial $p(x) = p_0 + \cdots + p_{m-1}x^{m-1} + x^m$*

of degree m we define the companion matrix of $p(x)$ as:

$$C(p(x)) = \begin{pmatrix} 0 & 0 & \dots & 0 & 0 & -p_0 \\ 1 & 0 & \dots & 0 & 0 & -p_1 \\ 0 & 1 & \dots & 0 & 0 & -p_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & 0 & -p_{m-1} \\ 0 & 0 & \dots & 0 & 1 & -p_m \end{pmatrix}$$

Corollary 1. *Given a primitive polynomial $p(x) \in \mathbb{F}_2[x]$ of degree m, the determinant of $C(p(x))$ over $\mathbb{F}_2$ is 1.*

Definition 31 (Circulant matrix)**.** *A circulant matrix $C([a_0, a_1, a_2, ..., a_{m-1}])$ of size $m \times m$ is a matrix where each of its rows is a right cyclic shift of the row above it. Hence, it has the following form:*

$$C([a_0, a_1, a_2, ..., a_{m-1}]) = \begin{pmatrix} a_0 & a_1 & a_2 & \dots & a_{m-2} & a_{m-1} \\ a_{m-1} & a_0 & a_1 & \dots & a_{m-3} & a_{m-2} \\ a_{m-2} & a_{m-1} & a_0 & \dots & a_{m-4} & a_{m-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ a_2 & a_3 & a_4 & \dots & a_0 & a_1 \\ a_1 & a_2 & a_3 & \dots & a_{m-1} & a_0 \end{pmatrix}$$

5.1 Construction Based on Companion Matrices

Theorem 6 ([CCR13])**.** *Let C be the companion matrix of a primitive polynomial $p(x) \in \mathbb{F}_q[x]$ with degree b. If $\alpha \in \mathbb{F}_{q^b}$ is a primitive element, then the map $\psi : \mathbb{F}_{q^b} \mapsto \mathbb{F}_q[C]$ such that $\psi(\alpha) = C$ is a Field isomorphism.*

Theorem 7 ([CCR13])**.** *Let C be the companion matrix of a primitive polynomial $p(x) \in \mathbb{F}_q[x]$ with degree b, and consider the field isomorphism $\psi : \mathbb{F}_{q^b} \mapsto \mathbb{F}_q[C]$ such that $\psi(\alpha) = C$*

where $\alpha \in \mathbb{F}_{q^b}$ is a primitive element. Then the map $\phi : Mat_{m\times t}(\mathbb{F}_{q^b}) \mapsto Mat_{m\times t}(\mathbb{F}_q[C])$ given by

$$\phi([\alpha_{ij}]) = [\psi(\alpha_{ij})]$$

is a ring isomorphism.

Theorem 8 ([CCR13])**.** *If $A \in Mat_{m\times t}(\mathbb{F}_{q^b})$ is a superregular matrix (Def. 17), then $G = \begin{bmatrix} I_{mb} & \phi(A) \end{bmatrix}$ is the generator of an MDS $\mathbb{F}_q$-linear code $\mathcal{C}_{\mathbb{F}_q^b}$ with parameters $[m+t, m, t+1]$.*

Example 5. *Consider the primitive polynomial $p(x) = 1 + x^2 + x^3 \in \mathbb{F}_2[x]$. Then,*

$$C(p(x)) = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \end{bmatrix}$$

Let $\alpha \in F_{2^3}$ be a primitive element and consider the superregular matrix

$$A = \begin{bmatrix} \alpha & 1 \\ 1 & \alpha^2 \end{bmatrix} \in Mat_{2\times 2}(F_{2^3})$$

Then $\phi(A) = \begin{bmatrix} C & I \\ I & C^2 \end{bmatrix} \in Mat_{2\times 2}(F_C)$ is a superregular 3-block matrix. Consequently,

$$G = \left[\begin{array}{c|cc} & C & I_3 \\ I_6 & & \\ & I_3 & C^2 \\ & & \end{array}\right]$$

is a generator matrix for an MDS $\mathbb{F}_2$-linear code $\mathcal{C}_{\mathbb{F}_2^3}$ with parameters $[4, 2, 3]$.

5.2 Construction Based on Circulant and Companion Matrices

Theorem 9. *The determinant of the binary circulant matrix $A = C([a, a, \ldots, a, b, b, \ldots, b])$ (i.e., with a repeated k times and b repeated $m-k$ times) of size $m \times m$ is given by*

$$\det(A) = \begin{cases} 0, & \text{if } \gcd(k, m) \neq 1 \\ [ka + (m-k)b](a-b)^{m-1}, & \text{otherwise} \end{cases}$$

Corollary 2. *The determinant of the binary circulant matrix $A = C([1, 1, \ldots, 1, 0, 0, \ldots, 0])$ (i.e., with 1 repeated k times and 0 repeated $m-k$ times) of size $m \times m$ is given by*

$$\det(A) = \begin{cases} 0, & \text{if } \gcd(k, m) \neq 1 \\ k, & \text{otherwise} \end{cases}$$

Corollary 3. *If $m > 2$ is a prime integer, then the determinant of the binary circulant matrix $A = C([1, 1, \ldots, 1, 0, 0, \ldots, 0])$ (i.e., with 1 repeated k times and 0 repeated $m-k$ times) over $\mathbb{F}_2$ is given by*

$$\det(A) = \begin{cases} 0, & \text{if } \gcd(k, m) \neq 1 \\ 1, & \text{otherwise} \end{cases}$$

We will denote by $C(k, m)$ the circulant matrix $C([1, 1, \ldots, 1, 0, 0, \ldots, 0])$ (i.e., with 1 repeated k times and 0 repeated $m-k$ times).

Theorem 10 ([CCR13])**.** *Let m be an odd prime number, k an odd positive integer and $p(x)$ a primitive polynomial of degree m. The matrix $G = [I_{2m}, A]$ with*

$$A = \begin{bmatrix} C(k, m) & C(k, m) \\ C(k, m) & C(p(x)) \end{bmatrix}$$

or

$$A = \begin{bmatrix} C(p(x)) & C(p(x)) \\ C(p(x)) & C(k,m) \end{bmatrix}$$

is the generator matrix of a MDS $\mathbb{F}_q$*-linear code with parameters* $[4m, 2m]$.

Definition 32 (Cyclotomic polynomial)**.** *The cyclotomic polynomial for* m *prime is given by* $\Phi_m(x) = \sum_{m-1}^{i=0} x^i$.

Theorem 11 ([CCR13])**.** *Consider the polynomial* $q(x) \in \mathbb{F}_2[x]$ *with degree less or equal than* m*. If* $\gcd(q(x), \Phi_m(x)) \neq 1$ *over* $\mathbb{F}_2[x]$*, then the determinant of the matrix* $C^\circ(q(x), m)$ *is zero over* $\mathbb{F}_2$*, where* $C^\circ(q(x), m) = C([q[x]] \parallel [0, \ldots, 0]_{m-\deg q(x)})$.

Theorem 12 ([CCR13])**.** *Let r(x) be a primitive polynomial with degree* $t < m$ *and* m *be a prime integer. If* $\gcd(\Phi_m(x), r(x)) = 1$ *then* $\det C^\circ(r(x), m) = 1$.

Theorem 13 ([CCR13])**.** *If* $m = 2^L$ *and* $p(x) = p_0 + p_1 x + \cdots + p_{m-1}x^{m-1} + x^m$ *is a irreducible polynomial, then* $\det(C([p_0, p_1, \ldots, p_{m-1}])) = 1$.

Theorem 14 ([CCR13])**.** *Let* $p(x)$ *be a primitive polynomial of degree* m *and* $r(x)$ *as before. The matrix* $G = [I_{2^m}, A]$ *with*

$$A = \begin{bmatrix} C^\circ(r(x), m) & C^\circ(r(x), m) \\ C^\circ(r(x), m) & C(p(x)) \end{bmatrix}$$

or

$$A = \begin{bmatrix} C(p(x)) & C(p(x)) \\ C(p(x)) & C^\circ(r(x), m) \end{bmatrix}$$

is the generator matrix of a MDS $\mathbb{F}_q$*-linear code with parameters* $[4m, 2m]$.

6 Conclusion and Future Work

In this paper, a detailed analysis of the MDS matrices application in cryptography was carried out. In addition to the analysis of MDS layer's property of perfect diffusion, it was shown in Theorem 4 that the usage of MDS matrices is a necessary (but not sufficient) condition for the non-existence of infinitely long invariant subspace trails for P-SPN schemes at the linear layer of inactive S-boxes rounds.

It was also explored in Section 5 several techniques to building MDS matrices based on companion and circulant matrices.

However the construction of MDS matrices that are sufficient itself to avoid infinitely long iterative subspaces trails still an open problem yet [GRS20], a possible future research topic may be the adaptation of the MDS constructions presented in Section 5 (proposed in [CCR13]) to satisfy the conditions of Theorem 3 aiming to avoid the existence of infinitely long invariant subspace trails.

Note that the core of Theorem 3 is to avoid the invariant action of the linear layer M, i.e. $S = MS$ for a certain subspace S. A possible approach to this problem might be to investigate the characteristics of the algebraic closure of M. Another approach, since S depends on M (see Definition 27), may be the investigation of the geometric properties of M as a linear transformation.

Let B_S be the set of vector in S that form a basis for this subspace. Let $\text{vol}(S)$ be the determinant of the matrix generated by placing in rows the vectors of B_S. We define $\overrightarrow{c} = \frac{\text{vol}(S)}{|S|}\sum_{s \in S} s$, i.e. the central vector of subspace S times the subspace volume.

Since any changes in S (in at least one vector) can be detected by modifications in the subspace center and/or volume, the vector $\overrightarrow{c}$ could be used to characterize the geometric behavior of a linear transformation M.

Figures 6 and 6 illustrates, respectively, how $\overrightarrow{c}$ should be on a 2-dimensional S toy model; and how the variance of S over a M linear transformation can be visualized by

mutations on $\overrightarrow{c}$, as proposed.

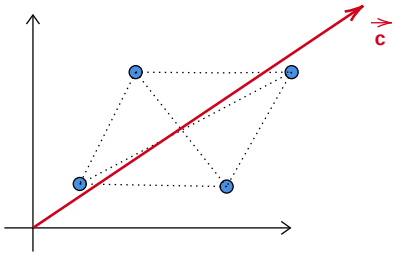

Figure 5: Vector $\mathbf{c}$ for some S.

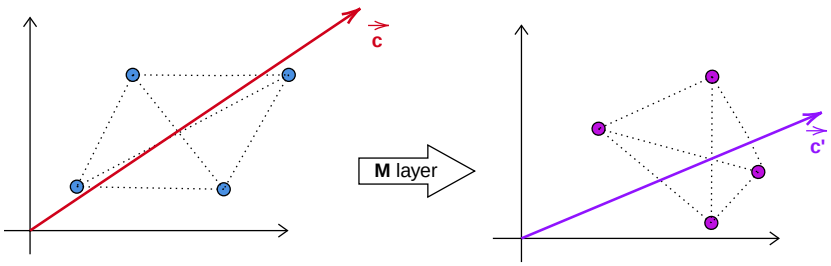

Figure 6: Instance of $S \neq MS$, leading to $\mathbf{c} \neq \mathbf{c}'$, for some S.

Acknowledgments

This work would not have been possible without the help of my advisor, Professor Ricardo Dahab (University of Campinas - Unicamp, Brazil); Professor Daniel Panario (Carleton University, Canada); and Professor Sara Cardell (Federal University of ABC - UFABC, Brazil). We spend valuable hours in study group sessions on MDS matrices.

Furthermore, I would like to everyone who helped, help and will help me during my journey; because the laurels of success shine more with your presence in my life.

References

[ARS+16] Martin Albrecht, Christian Rechberger, Thomas Schneider, Tyge Tiessen, and Michael Zohner. Ciphers for mpc and fhe. Cryptology ePrint Archive, Report 2016/687, 2016. https://eprint.iacr.org/2016/687.

[BBS99] Eli Biham, Alex Biryukov, and Adi Shamir. Cryptanalysis of skipjack reduced to 31 rounds using impossible differentials. In Advances in Cryptology — EUROCRYPT '99, pages 12–23. Springer Berlin Heidelberg, 1999.

[BLN16] Céline Blondeau, Gregor Leander, and Kaisa Nyberg. Differential-linear cryptanalysis revisited. Journal of Cryptology, 30(3):859–888, October 2016.

[BS91] Eli Biham and Adi Shamir. Differential cryptanalysis of DES-like cryptosystems. Journal of Cryptology, 4(1):3–72, January 1991.

[CCR13] Sara Cardell, Joan-Josep Climent, and Verónica Requena. A construction of mds array codes. volume 45, pages 47–58, 05 2013.

[DH76] W. Diffie and M. Hellman. New directions in cryptography. IEEE Transactions on Information Theory, 22(6):644–654, 1976.

[DR02] Joan Daemen and Vincent Rijmen. The Design of Rijndael: AES - The Advanced Encryption Standard. Information Security and Cryptography. Springer-Verlag Berlin Heidelberg, 2002.

[GGNPS13] B. Gérard, Vincent Grosso, M. Naya-Plasencia, and François-Xavier Standaert. Block ciphers that are easier to mask: How far can we go? In Guido Bertoni and Jean-Sébastien Coron, editors, Cryptographic Hardware and Embedded Systems - CHES 2013, pages 383–399, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[GKR+19] Lorenzo Grassi, Dmitry Khovratovich, Christian Rechberger, Arnab Roy, and Markus Schofnegger. Poseidon: A new hash function for zero-knowledge proof systems. Cryptology ePrint Archive, Report 2019/458, 2019. https://eprint.iacr.org/2019/458.

[GLR+19] Lorenzo Grassi, Reinhard Lüftenegger, Christian Rechberger, Dragos Rotaru, and Markus Schofnegger. On a generalization of substitution-permutation networks: The hades design strategy. Cryptology ePrint Archive, Report 2019/1107, 2019. https://eprint.iacr.org/2019/1107.

[GRR16] Lorenzo Grassi, Christian Rechberger, and Sondre Rønjom. Subspace trail cryptanalysis and its applications to aes. Cryptology ePrint Archive, Report 2016/592, 2016. https://eprint.iacr.org/2016/592.

[GRS20] Lorenzo Grassi, Christian Rechberger, and Markus Schofnegger. Weak linear layers in word-oriented partial spn and hades-like schemes. Cryptology ePrint Archive, Report 2020/500, 2020. https://eprint.iacr.org/2020/500.

[HP03] W. Cary Huffman and Vera Pless. Fundamentals of Error-Correcting Codes. Cambridge University Press, New York, NY, 2003.

[Knu95] Lars R. Knudsen. Truncated and higher order differentials. In Fast Software Encryption, pages 196–211. Springer Berlin Heidelberg, 1995.

[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.

[LTW18] Gregor Leander, Cihangir Tezcan, and Friedrich Wiemer. Searching for subspace trails and truncated differentials. IACR Transactions on Symmetric Cryptology, pages 74–100, March 2018.

[Mer90] Ralph C. Merkle. A certified digital signature. In Gilles Brassard, editor, Advances in Cryptology — CRYPTO' 89 Proceedings, pages 218–238, New York, NY, 1990. Springer New York.

[MM00] Ueli Maurer and James Massey. Local randomness in pseudo-random sequences. 06 2000.

[MS74] George Marsaglia and George P. H. Styan. Equalities and inequalities for ranks of matrices. Linear and Multilinear Algebra, 2(3):269–292, 1974.

[NIS01] National Institute of Standards and Technology. Announcing the Advanced Encryption Standard (AES), 2001. Federal Information Processing Standards Publication 197, http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf.

[RSA78] R. L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. Commun. ACM, 21(2):120–126, February 1978.

[Sha49] C. E. Shannon. Communication theory of secrecy systems. The Bell System Technical Journal, 28(4):656–715, 1949.

[Sti04] Douglas R. Stinson. Combinatorial Designs: Constructions and Analysis. Springer, New York, NY, 2004.

[Vau95] Serge Vaudenay. On the need for multipermutations: Cryptanalysis of MD4 and SAFER. In G. Goos, J. Hartmanis, J. Leeuwen, and Bart Preneel, editors, Fast Software Encryption, volume 1008, pages 286–297. Springer Berlin Heidelberg, Berlin, Heidelberg, 1995.