# Weak Linear Layers in Word-Oriented Partial SPN and HADES-Like Schemes

Lorenzo Grassi[1,2], Christian Rechberger[1] and Markus Schofnegger[1]

[1] IAIK, Graz University of Technology
[2] Digital Security Group, Radboud University, Nijmegen
firstname.lastname@iaik.tugraz.at
l.grassi@cs.ru.nl

**Abstract.** Designing cryptographic permutations and ciphers using a substitution-permutation network (SPN) approach where the nonlinear part does not cover the full state has recently gained attention due to favourable implementation characteristics in various scenarios.

For these word-oriented partial SPN schemes with a fixed linear layer, our goal is to better understand linear layer construction. In this paper we derive conditions which allow either to set up or to prevent attacks based on infinitely long truncated differentials with probability 1. Our analysis is rather broad as in contrast to earlier independent work on this problem, we consider *(1)* trails that are invariant and trails that are not, and *(2)* trails with and without active S-boxes.

In both cases (namely, active and inactive S-boxes), we are able to provide rigorous sufficient and necessary conditions that prevent the analyzed attacks. On the practical side, we present a tool which is able to determine whether a given linear layer is vulnerable based on these results.

Besides P-SPN schemes, our observations may also have a crucial impact on the very recent Hades design strategy, which mixes rounds with full S-box layers and rounds with partial S-box layers.

**Keywords:** Partial SPN · Linear Layer · Subspace Trails · Hades Schemes

## Contents

# 1 Introduction

Modern cryptography developed many techniques that go well beyond solving traditional confidentiality and authenticity problems in two-party communication. Among many others, this includes practical applications of secure multi-party computation (MPC), fully homomorphic encryption (FHE), and zero-knowledge proofs (ZK) that use symmetric primitives. A possible guiding principle for designs aiming at such applications is as follows: Linear operations are much cheaper to compute than nonlinear ones. This fact is also true in the context of masking, a widespread countermeasure against side-channel attacks (SCA) in which all the computations are performed on shared secrets[1].

Driven by all these application areas and settings, many new symmetric primitives have recently been proposed to reduce the multiplicative complexity in various ways. They include masking-friendly designs like PICARO [42], Zorro [24], LS-designs [31], several FHE-friendly symmetric encryption schemes such as LowMC [5], FLIP [41], Kreyvium [17] and Rasta [22], some MPC-friendly block ciphers such as MiMC [4, 30], GMiMC [3] and HadesMiMC [27] (and its hash variant Poseidon [25]), and some primitives dedicated to proof systems such as Jarvis and Friday [7], *Vision* and *Rescue* [6].

By minimizing the multiplicative complexity, these new schemes based on specialized designs outperform “classical” schemes when targeting these particular applications. At the same time, all these primitives are based on innovative constructions which are (in general) not well analyzed yet: In some cases, the implementation constraints at the base of their designs may have introduced some unexpected weaknesses. This was indeed the case for Zorro (broken by statistical attacks [45]), LowMC (first version broken by a higher-order differential attack [21]), Jarvis and Friday (broken by a Gröbner basis attack [2]), and more recently MiMC (broken by a higher-order differential attack [23]) and Starkad (only in the case in which the linear layer has a low multiplicative order [13]).

In this paper, we focus on word-oriented partial SPN and Hades-like schemes. Our goal is to better understand how the choice of the linear layer influences their security against some particular attacks explained in the following.

## 1.1 Our Contribution

A partial substitution-permutation network (P-SPN) is a variation of the SPN approach in which part of the substitution layer is replaced with an identity mapping (with the goal to decrease the overall number of nonlinear operations). As already mentioned, two concrete examples of P-SPN ciphers are Zorro [24] and LowMC [5]. Zorro is a 128-bit lightweight AES-like cipher which reduces the number of S-boxes per round from 16 to only 4 (to compensate, the number of rounds has been increased to 24), while LowMC is a flexible block cipher based on an SPN structure, which combines an incomplete S-box layer with a strong linear layer in order to provide security.

Arguably, the main difference between these two designs regards the choice of the linear layer. While Zorro uses the same linear layer in all rounds, LowMC uses different pseudo-random linear layers for each round. Even if the second strategy can potentially prevent statistical attacks (as discussed in [5]), it has some drawbacks. First of all, the implementation cost in terms of computation time or memory may become a problem, even when considering the optimizations proposed in [35, 20]. Moreover, the security analysis against other attacks may become more complicated, since the linear layer is different in each round. Finally, a poor choice of the linear layers may not provide security against statistical attacks, as shown concretely in [21].

For all these reasons, in this paper we focus only on the first strategy (which is also used in Hades-like schemes, discussed later): Our goal is to better understand *which*

[1] We recall that from a theoretical point of view, the problem of masking a cryptographic implementation has strong connections with the problem of secure multiparty computation [34, 32].

*properties a linear layer has to fulfill in order to prevent the existence of infinitely long truncated differentials with prob. 1* [37], or equivalently *infinitely long subspace trails* [28, 29] (namely, the existence of a non-trivial subspace $\mathcal{U} \subseteq \mathbb{F}^t$ of inputs that is mapped into a proper (affine) subspace of the state space over any number of rounds).

We consider both invariant and non-invariant subspaces, and we also provide results in the case of subspace trails with active S-boxes. In addition, we present an algorithm and a concrete implemented tool which, given a square matrix, can be used to detect infinitely long subspace trails in the case of inactive and active S-boxes.

**Influence of the Branch Number.** Let us focus on a word-oriented partial SPN scheme over $\mathbb{F}^t$, where the linear layer is simply defined as the multiplication with a $t \times t$ MDS matrix. Since such a matrix provides full diffusion at word level, and since a partial nonlinear layer is applied, one may expect that after a certain – even huge – number of rounds, the corresponding cipher is secure.

As we are going to show with a concrete example, this is not always the case. Indeed, consider a partial SPN scheme defined over $(\mathbb{F}_p)^4$ for some prime number $p$, and let the round transformation be

$$R^{(i)}\left(x = (x[1], x[2], x[3], x[4])^T\right) = k^{(i)} + \begin{pmatrix} 4 & 4 & 5 & 1 \\ 1 & 3 & 5 & 3 \\ 3 & 2 & 4 & 1 \\ 4 & 1 & 4 & 4 \end{pmatrix} \cdot \begin{pmatrix} S(x[1]) \\ x[2] \\ x[3] \\ x[4] \end{pmatrix} \tag{1}$$

for a "good" S-box $S : \mathbb{F} \to \mathbb{F}$, where $R^{(i)}$ denotes the $i$-th round function and $k^{(i)}$ denotes the $i$-th round key. Even though the $4 \times 4$ matrix used in this scheme can be an MDS matrix for sufficiently large[2] $p$, an invariant subspace trail generated by the subspace $\mathcal{S} = \left\langle (0, 1, -1, 1)^T \right\rangle$ can be set up for an *arbitrary* number of rounds.

**Infinitely Long Subspace Trails for Word-Oriented P-SPN Schemes.** The previous example allows us to conclude that a high branch number alone is not sufficient in the case of word-oriented partial SPN schemes when compared to the case of (full) SPN schemes. For this reason, in the following we analyze how the details of the matrix that defines the linear layer influences the security against statistical attacks. Specifically, working independently of the details of the S-box and of the values of the round keys and constants, we present sufficient and necessary conditions that allow to determine if a given matrix provides security w.r.t. the considered attacks (i.e., if infinitely long (non-trivial) subspace trails for the given matrix exist).

Both in the case of inactive S-boxes (see Section 3 for details) and in the case of active S-boxes (see Section 5 for details),

*(1)* we show that an infinitely long subspace trail exists if and only if the invertible fixed matrix that defines the linear layer satisfies some particular properties, and

*(2)* we show how to construct such an infinitely long subspace trail if it exists.

Moreover, if the matrix is diagonalizable, we show that the infinitely long subspace trail (if existent) is always related to the eigenspaces of the matrix. We further emphasize that we do not only focus on invariant subspace trails (in other words, a non-trivial infinitely long subspace trail is not necessarily invariant). Indeed, such a subspace trail is

*(1)* invariant if it is related to the eigenspaces of $M$, and

*(2)* not invariant if it is related to the eigenspaces of $M^k$ for $k \geq 2$.

[2] It is an MDS matrix for e.g. $p = 4\,206\,590\,407$, which results in a block size of approximately 128 bits.

In both cases, examples are provided to present and support the results. We remark that we do not impose any condition on the matrix $M$ (with the only exception that it is invertible), i.e., we do not limit ourselves to work only with MDS matrices. Moreover, the results we obtain are quite different from what is known for the SPN case.

To summarize, both in the case of active and inactive S-boxes, we present a rigorous *necessary and sufficient condition that guarantees that no infinitely long (invariant and iterative) subspace trail exists.*

**Dedicated Tool.** Together with our theoretical observations, we also provide practical Sage implementations based on our results. Given a square matrix, the tool and the underlying algorithms are able to detect the structural vulnerabilities described in this paper (invariant and iterative trails), both in the case of inactive and active S-boxes and for binary and prime fields.

The tool is split into three different algorithms to cover all our results. The vulnerability of a single matrix can be evaluated quickly, and to also get a better understanding of the percentage of vulnerable matrices for given matrix dimensions and field sizes, we applied our tool to large sets of pseudo-randomly sampled matrices. The result of these tests is that indeed the number of vulnerable matrices is significant, and sometimes even larger than 10%. All details about the tool and the results are given in Section 4 and Section 6.

**Impact on Hades-Like Schemes.** Our results have a considerable impact on the Hades strategy as well [27], recently proposed at Eurocrypt'20. This strategy is a high-level design approach for cryptographic permutations and keyed permutations addressing the needs of new applications that emphasize the role of multiplications in these designs, with a focus on simple arguments for its security. The main ingredient of the Hades strategy is to mix rounds with full S-box layers and rounds with partial S-box layers in order to provide good performance while still being secure. The external rounds with full S-box layers together with the wide trail strategy are used for the security against differential and linear attacks. The main goal of the middle rounds with a single S-box each is to provide security against algebraic attacks by increasing the degree of the overall scheme.

In [27], the authors define the linear layer as a multiplication of the state with a fixed MDS matrix (namely, a matrix with maximum branch number), and no other properties have to be fulfilled by the linear layer. It follows that in the case of a "weak" MDS matrix (namely, a matrix that does not satisfy the properties proposed in this work), an attacker can potentially choose an input space of texts for which no S-box is activated over all rounds with partial S-box layers. In such a case, the security of the corresponding design may potentially be lower. Indeed, if no S-box is active, the degree of the function does not increase in the rounds with partial S-box layers when working with these chosen texts. Consequently, algebraic attacks become possible, as demonstrated in [13], where preimage attacks against the sponge hash function Starkad (whose linear layer has a low multiplicative order) are proposed.

At the same time, a "strong" linear layer can be used by the designer in order to increase the security against statistical attacks by exploiting the presence of rounds with partial S-box layers [36]. For all these reasons, we suggest[3] that the MDS matrix defining the linear layer for such a scheme must not be "weak" with respect to the properties given in this paper. We also point out that currently there is no known key-recovery attack on HadesMiMC exploiting these properties.

## 1.2 Related Work

**Relation between Eigenvalues, Eigenvectors, and Invariant Subspace Trails.** The relation between the eigenvalues and eigenvectors of the linear layer matrix and the existence

[3] This is also supported by the designers of Hades (private communication).

of an infinitely long (invariant) subspace trail is already known in the literature. Such a relation was pointed out by Abdelraheem et al. [1], and later on generalized by Beyne in [12]. In more detail, Abdelraheem et al. found such a result by analyzing the invariant subspace trails of PRINTcipher (which was presented one year before in [38]), while Beyne found such a result as a generalization and improvement of the nonlinear invariant subspace attack on Midori-64 [44]. In particular, in [12] *a connection between the eigenvalues of the correlation matrix that defines the round function and the existence of an invariant subspace trail* is made. More details are given in Appendix A.

The results presented in [1, 12] focus on SPN ciphers and on invariant subspaces only. As a consequence, this analysis heavily depends both on the effect of the key (namely, the invariant subspace only holds in the case of weak keys[4]) and in general on the details of the S-box. Here we point out that the situation for partial SPN ciphers/permutations is different: *The results found for SPN ciphers do not (trivially) apply to the P-SPN case and vice-versa.*

First of all, in P-SPN ciphers, it is possible to set up infinitely long invariant subspaces independently of the choice of the key, of the key schedule, of the round constants, and of the details of the S-box. In other words, for the case of P-SPN ciphers, the existence of an infinitely long invariant subspace trail may depend *only* on the properties of the linear layer, which is not the case for an SPN cipher due to the full nonlinear layer (since every round of an SPN cipher contains at least one active S-box, it is not possible to work independently of it).

Secondly, this also has an impact on the subspace trail that can be set up. For SPN ciphers, due to the restriction on the key and on the round constants, it is possible to set up an invariant subspace trail e.g. of the form $R(\mathcal{U} + v) = \mathcal{U} + w$ only in the case in which $v$ is in a subset of $\mathbb{F}^t$. This restriction is not necessary in the P-SPN case. Moreover, for this class of ciphers, the following facts hold:

- The subspace trail does not need to be invariant in order to be infinitely long (i.e., we do not restrict ourselves to the case $R(\mathcal{U} + v) = \mathcal{U} + w$).
- A non-trivial infinitely long invariant subspace trail can potentially be set up both in the case in which *no S-box is active*, and – for the first time – also in the case in which *some (or even all) S-boxes are active.* The crucial point is that we do not need to consider the details of the S-box (i.e., we do not require the S-box to fulfill any specific properties), which is not possible for the case of SPN ciphers.

More details about this are given in the following.

**Tools for (Invariant) Subspace Trails for SPN Ciphers.** Given SPN ciphers/schemes, tools that look for (finitely long) invariant subspace trails and/or (weak-key) subspace trails have already been published in the literature, see e.g. [39] and [26]. In both cases, the goal is to build the smaller subspace of texts that preserves some given properties (e.g., being invariant) for a certain number of rounds under some weak keys. The tools proposed in this paper (for the case of active S-boxes) resemble, in some ways, the strategies already proposed in [39, 26] in order to search for the infinitely long subspace trails we are looking for. Since the goals of these tools are not equal, some important differences can be highlighted. First, here we focus on the matrix that defines the linear layer, working independently of the round keys, the round constants, and the details of the S-box, while in [39, 26] these details are relevant. Secondly, the results presented in [39, 26] aim to cover a finite number of rounds, while here we focus on preventing infinitely long subspace trails. More details about this are given in Section 6.

[4] The existence of such an invariant subspace can be easily prevented by a careful choice of the round constants, as shown in [11].

**Infinitely Long Invariant Subspace Trails for Hades.** We note that the idea of considering infinitely long invariant subspace trails for a certain class of linear layer – that is, Cauchy matrices (a class of MDS matrices) over a Boolean field $\mathbb{F}_{2^n}$ generated in the very specific way given in [25] – has recently been studied independently in [36] and [13]. In there, the authors show that matrices belonging to this class always have a low multiplicative order[5]. This fact may introduce weaknesses in the scheme[6], due to the existence of infinitely long (non-trivial) subspace trails that can be exploited to break it. A concrete example of this is the preimage attack on STARKAD presented in [13].

While the observations presented in [36] and [25] focus on a small class of (Cauchy) matrices, our results do not make such specific assumptions about the matrices used in the linear layers. For example, this may be useful in the case in which one is interested in studying possible variants of HADES-like schemes in which the MDS matrix is replaced by a matrix with a smaller (known) branch number which is cheaper to implement (e.g., a near-MDS matrix). Moreover, let us recall that the middle (partial) rounds can potentially be exploited to increase the security against statistical attacks, as suggested in [36]. The results presented here are naturally relevant in such a case.

**Security against Statistical Attacks.** In this paper, we present properties which a matrix defining the linear layer *must not* satisfy in order to prevent infinitely long subspace trails. However, in general this does not help in predicting the number of rounds necessary to provide security against, for example, statistical attacks. Such a contribution can be found in [10], where the authors propose generic techniques for differential and linear cryptanalysis of SP networks with partial nonlinear layers. However, it does not analyze which properties a matrix must satisfy in order to prevent infinitely long subspace trails – as we do here. Hence, our work and the one proposed in [10] complement each other.

# 2 Preliminaries

**Notation.** We denote subspaces with calligraphic letters (e.g., $\mathcal{S}$). Further, we use the superscript notation together with parentheses to differentiate subspaces with similar properties (e.g., $\mathcal{S}^{(i)}$). By $\mathcal{S}^c$ we denote the complementary subspace of $\mathcal{S}$. We recall that two cosets $\mathcal{S}+a$ and $\mathcal{S}+b$ are considered to be equivalent if and only if $a-b \in \mathcal{S}$. Matrices are denoted by non-calligraphic letters, and the superscript notation for matrices is used to indicate powers of matrices in their traditional form. The entry of a vector $x \in \mathbb{F}^t$ is denoted by $x[i]$ for $i \in \{1, \ldots, t\}$, while the entry of a matrix $M$ in the $j$-th column of the $i$-th row is denoted either by $M_{i,j}$ or by $M[i,j]$. We denote by $\{e_1, \ldots, e_t\}$ the unit vectors of $\mathbb{F}^t$ (i.e., $e_i$ has a single 1 in the $i$-th word). Finally, given a generic subspace $\mathcal{X} \subseteq \mathbb{F}^t$ and a matrix $M$, let $M \cdot \mathcal{X} := \{M \cdot x \mid x \in \mathcal{X}\}$.

## 2.1 SPN and Partial SPN Schemes

In this paper, we will focus on partial SPN ciphers and permutations over $((\mathbb{F}_q)^t, +, \cdot)$, where $q \geq 2$ is a prime power (if $q = 2^n$, then $((\mathbb{F}_q)^t, +, \cdot)$ corresponds to $((\mathbb{F}_{2^n})^t, \oplus, \cdot)$). These schemes are similar to classical (full) SPN schemes, with the only difference being that the S-boxes (i.e., the nonlinear functions of the cipher) are not applied to the whole state.

Before going on, we highlight that all our results are independent of the round keys and constants. For this reason, in the following we do not clearly distinguish between ciphers and permutations, and we occasionally just refer to them using the term "schemes".

[5] The multiplicative order of a matrix $M$ is the smallest (integer positive) exponent $k \geq 1$ such that $M^k = \mu I$, where $\mu \in \mathbb{F}$ and $I$ is the identity matrix.

[6] In [36], the authors show how to fix this problem by choosing Cauchy matrices that do not have such properties and how to exploit them in order to provide stronger security arguments against statistical attacks.

**SPN Ciphers.** We denote the application of $r$ rounds of an SPN cipher by $E_k^r : \mathbb{F}^t \to \mathbb{F}^t$, where $k \in \mathbb{F}^t$ is a fixed secret key and $t \in \mathbb{N}$ denotes the number of cells. For every input $x = (x[1], \ldots, x[t]) \in \mathbb{F}^t$, the encryption is defined by $E_k^r(x) = (F_r \circ \cdots \circ F_0)\,(x + k^{(0)})$, where $F_i : \mathbb{F}^t \to \mathbb{F}^t$ is defined as $F_i(x) = R(x) + k^{(i)}$ for $i \in [1,t]$ and the round keys $k^{(0)}, \ldots, k^{(r)} \in \mathbb{F}^t$. 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., we have $R : \mathbb{F}^t \to \mathbb{F}^t$ with

$$R(x) = (M \circ S)(x) = M(S_1(x[1]), \ldots, S_t(x[t])),$$

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

$$M(x) = \begin{pmatrix} M_{1,1} & M_{1,2} & \ldots & M_{1,t} \\ M_{2,1} & M_{2,2} & \ldots & M_{2,t} \\ \vdots & \vdots & \ddots & \vdots \\ M_{t,1} & M_{t,2} & \ldots & M_{t,t} \end{pmatrix} \cdot \begin{pmatrix} x[1] \\ x[2] \\ \vdots \\ x[t] \end{pmatrix},$$

where $M_{i,j} \in \mathbb{F}$ for $i \in [1,t]$ and $j \in [1,t]$.

**Definition 1.** A linear layer $M \in \mathbb{F}^{t \times t}$ is *non-trivial* if it ensures full diffusion[7] (in the sense that each word of the output depends on each word of the input and vice versa) after a *finite* number of rounds.

Note that all SPN ciphers can be written in this way. Just to give some examples, if $M$ is an MDS matrix[8], the cipher is similar to SHARK [43]. For AES [19] or AES-like ciphers (where the linear layer is obtained as a combination of a ShiftRows and a MixColumns operation), many entries of $M$ are equal to 0.

**Partial SPN (P-SPN) Ciphers.** The main and only difference to an SPN cipher regards the S-box layer. For the case of partial SPN (P-SPN) ciphers, the round (and so the S-box layer) is defined as

$$R(\cdot) = M \circ \underbrace{(S_1 \mid\mid \cdots \mid\mid S_s \mid\mid I_{s+1} \mid\mid \cdots \mid\mid I_t)}_{\text{S-box layer}}(\cdot), \tag{2}$$

where $1 \leq s < t$ and where $I_{s+1} = \cdots = I_t$ are identity functions. In other words, instead of having a full S-box layer, the nonlinear functions are applied only to a part of the state, while the rest of the state remains unchanged.

In this paper, we assume that the $s$ S-boxes are applied to the first $s$ words. Note that given any partial SPN cipher, it is always possible to find an equivalent representation such that the S-boxes are applied to the first $s$ words.

**Hades-Like Schemes.** The recently proposed HADES strategy [27] combines both SPN and partial SPN schemes in the following way:

- The initial $R_f$ and the final $R_f$ rounds contain full S-box layers, for a total of $R_F = 2R_f$ rounds with full S-box layers.
- In the middle of the construction, $R_P$ rounds with partial S-box layers are used.

Roughly speaking, $R_F$ rounds provide security against statistical attacks, while $R_P$ rounds are exploited in order to increase the overall degree of the encryption/decryption function, in an attempt to provide security against algebraic attacks.

[7] The linear layer defined by the multiplication with $M$ provides full diffusion if there exists $r \in \mathbb{N}$ such that after the $r$-round permutation every output word $y_j$ depends on every input word $x_i$ for each state $x \in \mathbb{F}^t$, where $i \in [1,t]$ and $j \in [1,t]$. For example, the identity matrix does not fulfill this condition.

[8] A matrix $M \in \mathbb{F}^{t \times t}$ is called a maximum distance separable (MDS) matrix iff it has a branch number $\mathcal{B}(M)$ equal to $\mathcal{B}(M) = t+1$. The branch number of $M$ is defined as $\mathcal{B}(M) = \min_{x \in \mathbb{F}^t}\{\text{wt}(x) + \text{wt}(M(x))\}$, where $\text{wt}(\cdot)$ is the bundle weight in wide trail terminology.

## 2.2 Invariant Subspaces and Subspace Trails

### 2.2.1 Invariant Subspace Attack

The invariant subspace attack, introduced in [38] and reconsidered e.g. in [39], is based on the possibility to set up an invariant subspace trail, defined as follows.

**Definition 2.** Let $K_{\text{weak}}$ be a set of keys and $k \in K_{\text{weak}}$, with $k = \left(k^{(0)}, \ldots, k^{(r)}\right)$, where $k^{(j)}$ is the $j$-th round key. For $k \in K_{\text{weak}}$, 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 $i \in \{1, \ldots, r\}$ there exists a non-empty set $A_i \subseteq \mathcal{IS}^c$ (where $\cdot^c$ denotes the complement) for which

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

All keys in the set $K_{\text{weak}}$ are weak keys.

Let us remark the main difference for invariant subspace attacks when working with partial SPN ciphers instead of SPN ones. In this last case and to the best of our knowledge, the sets $A_i$ are (almost always) non-trivial subsets of $\mathbb{F}^t$. As shown in the following, this restriction is not mandatory in the case of partial SPN schemes. For this reason, in the following we work independently of the details of the S-box, and we assume that $A_i = \mathbb{F}^t$ for each $i$ and that the set $K_{\text{weak}}$ is equal to the set of all possible keys.

### 2.2.2 Subspace Trail Attack

Subspace trails were first defined in [28], and they are strictly related to truncated differential attacks, as shown in [40]. We refer to [28] for more details about the concept of subspace trails. However, our treatment here is meant to be self-contained.

**Definition 3.** 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 $i \in \{1, \ldots, r\}$ and for each $a_i$ there exists $a_{i+1} \in \mathcal{U}_{i+1}^c$ such that

$$R^{(i)}(\mathcal{U}_i + a_i) \subseteq \mathcal{U}_{i+1} + a_{i+1},$$

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 called a *constant-dimensional subspace trail.*

**Iterative (Constant-Dimensional) Subspace Trails.** We now introduce the concept of infinitely long iterative (constant-dimensional) subspace trails.

**Definition 4.** Let $\{\mathcal{V}_1, \mathcal{V}_2, \ldots, \mathcal{V}_r\}$ be a constant-dimensional subspace trail for $r$ rounds. We call this subspace trail an *infinitely long iterative (constant-dimensional) subspace trail of period* $r$ for the considered scheme if it repeats itself an infinite number of times, i.e., if

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

is an infinitely long subspace trail.

Clearly, an invariant subspace trail is also an iterative subspace trail for the case of P-SPN schemes (under the previous assumptions), 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 $\{\mathcal{V}_1, \ldots, \mathcal{V}_r\}$ be an infinitely long iterative subspace trail of period $r$. If $\dim(\langle \mathcal{V}_1, \ldots, \mathcal{V}_r \rangle) < t$, then $\langle \mathcal{V}_1, \ldots, \mathcal{V}_r \rangle$ generates an infinitely long invariant subspace trail.*

*Proof.* The subspace $\langle \mathcal{V}_1, \ldots, \mathcal{V}_r \rangle$ is invariant since each coset of $\mathcal{V}_i$ is mapped into a coset of $\mathcal{V}_{i+1}$ (where the subindex is taken modulo $r+1$). □

While, to the best of our knowledge, no example of infinitely long iterative constant-dimensional subspace trails for SPN ciphers is given in the literature, a poor choice of the linear layer allows to find them for the case of P-SPN schemes.

**Weak-Key Subspace Trails.** For completeness, we mention that a generalization of the two previous attacks, called "weak-key subspace trail attack", has been proposed in [26] (it basically corresponds to a subspace trail that holds for a class of weak keys only).

**Truncated Differential Trails.** Before going on, we briefly mention the link between truncated differential trails and subspace trails. Differential attacks [15] 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 [37], 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 (see e.g. [16] for details). 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:

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

The relation between truncated differential trails and subspace trails has been studied in details in [40, 16]. Finally, impossible differential and truncated impossible differential attacks based on differentials that hold with probability 0 have been studied in [14].

## 2.3 Preliminary Assumptions

Before going on, we make clear that in our work we consider the following assumptions.

**"Generic" S-Box.** We assume that the S-box has no linear structure (in other words, for an S-box $S$, it is not possible to find $\mathcal{U}, \mathcal{V} \subseteq \mathbb{F}$ s.t. for each $u$ there exists $v$ for which $S(\mathcal{U} + u) = \mathcal{V} + v$). Under this assumption, one can work independently of the details of the S-box. Indeed, as was shown in [40], there are only two essential subspace trails ($\{0\} \to \{0\}$ and $\mathbb{F} \to \mathbb{F}$) when working at word level if the S-box has no non-trivial linear structure. E.g., both the AES S-box and the cube one ($x \mapsto x^3$) satisfy this assumption.

**No Weak Keys.** We only consider infinitely long constant subspace trails which are independent of the key. E.g., we assume that the key schedule prevents the possibility of setting up infinitely long constant subspace trails for a class of weak keys.

**Limited Number of S-Boxes.** We further assume $s < \lceil t/2 \rceil$, i.e., that the number of S-boxes $s$ is smaller than $\lceil t/2 \rceil$. This implies that the choice of the linear layer is crucial to guarantee that at least one S-box is active after a *finite* number of rounds. Indeed, in the case in which a fixed linear layer matrix $M$ is used, let $2 \leq b \leq t+1$ be its branch number. If $2t - 2s < b$, then at least $b + 2s - 2t \geq 1$ S-boxes are active in every two consecutive rounds. Note that this can never happen if $s < \lceil t/2 \rceil$ (equivalently, $s \leq \lceil t/2 \rceil - 1$), since $2t - 2s \geq t + 2 > b$.

# 3 Subspace Trails for P-SPN Schemes (Inactive S-Boxes)

In the case of SPN ciphers, (weak-key) infinitely long subspace trails can be prevented by carefully choosing the round constants (see [11] for details) and by exploiting the fact that a full S-box layer together with a reasonable linear layer provides full diffusion after a finite number of rounds. In the case of P-SPN schemes, however, the situation is different.

First of all, due to the fact that the S-box layer is not complete, the details of the round constants (together with a non-trivial linear layer) are not sufficient by themselves to provide security against the infinitely long subspace attacks just recalled. In this sense, the linear layer plays a crucial role in order to provide security. Here we focus on constructions in which the same linear layer is used in each round (e.g., `Zorro` [24]).

## 3.1 Preliminary Results

Due to the fact that the nonlinear layer is only partial in P-SPN schemes, parts of the state go through the S-box layer unchanged. In particular, if the nonlinear layer consists of $s \geq 1$ S-boxes and $t - s \geq 1$ identity functions, 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 the $s$ S-boxes are applied to the first $s$ words and by choosing texts in the same coset of $\mathcal{S} = \langle v_1, \ldots, v_d \rangle$ (where $d = \dim(\mathcal{S}) \geq t - s \cdot \left\lfloor \frac{t-s}{s} \right\rfloor$) such that

$$\forall i \in \{1, \ldots, \lfloor (t-s)/s \rfloor\}, \forall j \in \{1, \ldots, d\} : \quad (M^{i-1} \cdot v_j)[1, 2, \ldots, s] = 0 \mid\mid 0 \mid\mid \cdots \mid\mid 0 \in \mathbb{F}^s,$$

where $M^0 = I$ is the identity matrix, it follows that no S-box is active in the first $\left\lfloor \frac{t-s}{s} \right\rfloor$ rounds. We formalize this result in the following definition.

**Definition 5.** Consider the case of a P-SPN scheme over $\mathbb{F}^t$ with $1 \leq s < t$ S-boxes applied to the first $s$ words defined as in Eq. (2). Let $\mathcal{S}^{(i)}$ be defined as

$$\mathcal{S}^{(i)} = \left\{ v \in \mathbb{F}^t \;\middle|\; (M^j \cdot v)[1, \ldots, s] = 0 \mid\mid \cdots \mid\mid 0 \in \mathbb{F}^s, j < i \right\}, \tag{3}$$

where $\mathcal{S}^{(0)} = \mathbb{F}^t$, and where $\dim\left(\mathcal{S}^{(i)}\right) \geq t - i \cdot s$. Then $\mathcal{S}^{(i)}$ generates a subspace trail for the first $i$ (consecutive) rounds with no active S-boxes. Further, note that $\mathcal{S}^{(i+1)} \subseteq \mathcal{S}^{(i)}$.

**Lemma 1.** *Given a P-SPN scheme over $\mathbb{F}^t$ with $s$ S-boxes applied to the first $s$ words defined as in Eq.* (2)*, let $\mathcal{S}^{(i)}$ be defined as in Definition 5. Then, for each $i \geq 1$,*

$$\mathcal{S}^{(i+1)} = \left\{ v \in \mathcal{S}^{(i)} \;\middle|\; (M \cdot v)[1, \ldots, s] = 0 \mid\mid \cdots \mid\mid 0 \in \mathbb{F}^s \right\} = \mathcal{S}^{(i)} \cap (M^{-(i-1)} \cdot \mathcal{S}^{(1)}) \subseteq \mathcal{S}^{(i)}.$$

*Proof.* Given $\mathcal{S}^{(1)} = \langle e_{s+1}, \ldots, e_t \rangle$, note that $(M \cdot x)[1, \ldots, s] = 0 \mid\mid \cdots \mid\mid 0 \in \mathbb{F}^s$ if and only if $M \cdot x \in \langle e_{s+1}, e_{s+2}, \ldots, e_t \rangle = \mathcal{S}^{(1)}$, or equivalently $x \in (M^{-1} \cdot \mathcal{S}^{(1)})$. Hence,

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

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 $x \in (M^{-(i-1)} \cdot \mathcal{S}^{(1)})$ □

In the case in which $\dim\left(\mathcal{S}^{\left(\left\lfloor \frac{t-s}{s} \right\rfloor\right)}\right) \geq s$, the previous definition can naturally be extended to more rounds, as stated in the following.

**Proposition 2.** *Consider the case of a P-SPN scheme over $\mathbb{F}^t$ with $1 \leq s < t$ S-boxes applied to the first $s$ words as in Eq.* (2)*, and let $\mathcal{S}^{(i)}$ be defined as before. Let $\mathfrak{R} \geq \left\lfloor \frac{t-s}{s} \right\rfloor$ s.t. $\dim\left(\mathcal{S}^{(\mathfrak{R})}\right) \geq 1$ and $\dim\left(\mathcal{S}^{(\mathfrak{R}+1)}\right) = 0$. For each $r \leq \mathfrak{R}$, the collection*

$$\left\{ \mathcal{S}^{(r)}, M \cdot \mathcal{S}^{(r)}, M^2 \cdot \mathcal{S}^{(r)}, \ldots, M^{r-1} \cdot \mathcal{S}^{(r)} \right\}$$

*is a subspace trail for the first $r$ rounds (with no active S-boxes).*

This well-known result (see e.g. [5, Sect. 5.1] or [24, Sect. 4.1]) does not require any assumption about the matrix $M$ that defines the linear layer. In the following, we will explore in which cases it is possible to set up an infinitely long subspace trail.

## 3.2 Infinitely Long Invariant Subspace Trails: A (Sufficient) Condition on the Linear Layer $M$

As is well-known in the literature (see e.g. the results presented in [1, 12] and recalled in Appendix A), one possible strategy to set up invariant subspace trails is to analyze the eigenspaces of the matrix $M$ that defines the linear layer. Here we exploit the same approach, but first we recall some preliminary concepts.

**Definition 6.** Given $M \in \mathbb{F}^{t\times t}$, the subspace $\mathcal{P} = \langle \rho_1, \ldots, \rho_d \rangle \in \mathbb{F}^t$ is the (right) eigenspace of $M$ for the eigenvalue $\lambda$ if the condition $M \cdot \rho_i = \lambda \cdot \rho_i$ is satisfied $\forall i \in \{1, \ldots, d\}$.

**Definition 7.** $M$ is a diagonalizable matrix[9] if and only if there exists an (invertible) matrix $P \in \mathbb{F}^{t\times t}$ s.t. $P^{-1} \cdot M \cdot P = D = \text{diag}(\lambda_1, \ldots, \lambda_t)$ is a diagonal matrix.

**Theorem 1.** *Given a P-SPN scheme with $s$ S-boxes per round defined as in Eq. (2), let $M \in \mathbb{F}^{t\times t}$ be an invertible matrix. Let $\lambda_1, \ldots, \lambda_\tau$ be its eigenvalues and let $\mathcal{P}_1, \ldots, \mathcal{P}_\tau$ be the corresponding eigenspaces. Let*

$$\mathcal{IS} = \langle \mathcal{P}_1 \cap \langle e_{s+1}, \ldots, e_t \rangle, \ldots, \mathcal{P}_\tau \cap \langle e_{s+1}, \ldots, e_t \rangle \rangle .$$

*If $1 \leq \dim(\mathcal{IS}) < t$, then $\mathcal{IS}$ generates a (non-trivial) infinitely long invariant subspace trail (with no active S-boxes).*

Equivalently, let $\mathcal{IS}$ be defined as $\mathcal{IS} = \langle \mathcal{P}'_1, \ldots, \mathcal{P}'_\tau \rangle$ , where $\mathcal{P}'_i \subseteq \mathcal{P}_i$ is a subspace of $\mathcal{P}_i$ for $i \in \{1, \ldots, \tau\}$. If $\mathcal{IS} \cap \langle e_{s+1}, \ldots, e_t \rangle = \mathcal{IS}$, it generates an infinitely long invariant subspace trail. This equivalent definition will be used in the following, and we emphasize that this result provides only a *sufficient* condition.

*Proof.* To prove the previous result, we have to show that for each $a \in \mathbb{F}^t$ there exists $b$ s.t. $M \circ S(\mathcal{IS} + a) = \mathcal{IS} + b$. Hence, we omit the key and constant additions since they only change the coset.

First of all, note that no S-box is active since $\mathcal{IS} \subseteq \langle e_{s+1}, \ldots, e_t \rangle$. Hence, only the coset changes through the S-box layer. Secondly, since $\mathcal{P}_i$ is an eigenspace of the linear layer $M$ for each $i \in \{1, \ldots, \tau\}$, it follows that $\mathcal{P}_i \cap \langle e_{s+1}, \ldots, e_t \rangle$ remains invariant through it. The result follows immediately. □

It is crucial to work independently on the eigenspaces of $M$. Indeed, consider the case in which $\mathcal{P}_1 = \langle v \rangle$, $\mathcal{P}_2 = \langle w \rangle$, and $\langle \mathcal{P}_1, \mathcal{P}_2 \rangle \cap \langle e_{s+1}, \ldots, e_t \rangle = \langle v + \alpha w \rangle$. Given $x \in \langle \mathcal{P}_1, \mathcal{P}_2 \rangle \cap \langle e_{s+1}, \ldots, e_t \rangle$, $M \cdot x$ does not belong to such a subspace since $M \cdot (v + \alpha w) = \lambda_v \cdot \left( v + \alpha \cdot \frac{\lambda_w}{\lambda_v} \cdot w \right)$, where $\lambda_w \neq \lambda_v$.

**Example.** Consider the P-SPN scheme over $\mathbb{F}^4$ with $s = 1$ proposed in Eq. (1).

- In the case in which the $4 \times 4$ matrix $M$ is defined as in Eq. (1), $\mathcal{IS} = \langle (0, 1, -1, 1)^T \rangle$ generates an infinitely long invariant subspace trail. Indeed, note that $(0, 1, -1, -1)^T$ is an eigenvector of $M$ and that $\langle (0, 1, -1, 1)^T \rangle \cap \langle e_2, e_3, e_4 \rangle = \langle (0, 1, -1, 1)^T \rangle$ (hence, this is a concrete example of the result given in the previous theorem).
- If $M = \text{circ}(2, 3, 1, 1)$, the only eigenspaces are given by $\langle (1, 1, 1, 1)^T \rangle$ and $\langle (1, -1, 1, -1)^T \rangle$ (with eigenvalues equal to 7 and $-1$, respectively). They both do not satisfy the results of the theorem just given.

[9] A $t \times t$ matrix is diagonalizable if and only if the sum of the dimensions of its eigenspaces is equal to $t$.

## 3.3 Linear Layers with Low Multiplicative Order

As a next step, here we provide a first sufficient condition that, if satisfied by $M$, leads to an infinitely long iterative (non-invariant) subspace trail.

**Proposition 3.** *Given a P-SPN scheme over $\mathbb{F}^t$ defined as in Eq. (2), let $M \in \mathbb{F}^{t\times t}$ be an invertible matrix. If there exists $l \in \{2, \ldots, \mathfrak{R}\}$ (where $\mathfrak{R} \geq \left\lfloor \frac{t-s}{s} \right\rfloor$ is defined as in Proposition 2) and $\mu \in \mathbb{F} \setminus \{0\}$ such that $M^l = \mu \cdot I$ (equivalently, if $M$ has multiplicative order $l$), where $I \in \mathbb{F}^{t\times t}$ is the identity matrix, it is always possible to find an infinitely long iterative subspace trail.*

*Proof.* As we have seen before, it is always possible to find an initial subspace of the form $\left\{\mathcal{S}^{(l)}, M\cdot\mathcal{S}^{(l)}, M^2\cdot\mathcal{S}^{(l)}, \ldots, M^{l-1}\cdot\mathcal{S}^{(l)}\right\}$ such that no S-box is active for the first $l \leq \mathfrak{R}$ rounds (see Definition 5). Here, we only have to show that such an $l$-round subspace trail is repeated infinitely. To do this, we compute $M^i \cdot \mathcal{S}^{(l)}$ for $i \geq l$. By definition, there exist $j_1, j_2 \in \mathbb{N}$ s.t. $i = j_1 \cdot l + j_2$, where $j_2 < l$. Thus,

$$M^i \cdot \mathcal{S}^{(l)} = (M^l)^{j_1} \cdot M^{j_2} \cdot \mathcal{S}^{(l)} = (\mu \cdot I)^{j_1} \cdot M^{j_2} \cdot \mathcal{S}^{(l)} = M^{j_2} \cdot \mathcal{S}^{(l)}. \qquad \square$$

For example, $\left\{\mathcal{S}^{(l)}, M\cdot\mathcal{S}^{(l)}, M^2\cdot\mathcal{S}^{(l)}, \ldots, M^{l-1}\cdot\mathcal{S}^{(l)}\right\}$ is an infinitely long iterative subspace trail which is not invariant.

### Cauchy Matrices in [25]: A Concrete Example from the Literature

A concrete example has recently been pointed out by Keller et al. [36] and by Beyne et al. [13]. In these papers, the authors focus on the Cauchy matrix $\mathfrak{M} \in (\mathbb{F}_{2^n})^{t\times t}$ proposed in [25] and defined as

$$\mathfrak{M}_{i,j} = \frac{1}{x_i + x_j + r},$$

where $x_i = i - 1$ for $i \in \{1, \ldots, t\}$ and $t \leq r \leq p - t$. Such a matrix is used as the linear layer of some HADES-like permutations, namely STARKAD$^\pi$ and POSEIDON$^\pi$ [25].

In [46, Sect. 3.2] and in [36, 13], the authors prove that if $t = 2^\tau$, the previous matrix has a multiplicative complexity equal to 2, namely that $\mathfrak{M}^2$ is a multiple of the identity.[10] Hence, the previous result applies perfectly to this case.

## 3.4 Infinitely Long Iterative (Non-Invariant) Subspace Trails: A Sufficient Condition on the Linear Layer $M$

Until now, we focused only on the properties of $M$. However, since we are not working in a closed field, a possible generalization of the previous result can be presented.

Let $M$ be an invertible matrix in $\mathbb{F}^{t\times t}$. If $M$ is diagonalizable, then $M^l$, where $l \in \mathbb{N}$, is also diagonalizable:

$$P \cdot M \cdot P^{-1} = D \implies P \cdot M^l \cdot P^{-1} = D^l.$$

The other direction is not true in general, as given in the following proposition.

**Proposition 4** ([33]). *If $M$ is invertible, $\mathbb{F}$ is algebraically closed, and $M^l$ is diagonalizable for some $l$ that is not an integer multiple of the characteristic of $\mathbb{F}$, then $M$ is diagonalizable.*

Since no finite field can be algebraically closed, it follows that $M^l$ may contain more eigenvalues than $M$. In other words, if $\lambda$ is an eigenvalue of $M$, then $\lambda^l$ is also an eigenvalue

[10] In [13], the authors generalize the result by assuming that $\{x_1, x_2, \ldots, x_t\}$ forms a closed subgroup of $GF(2^n)$. By definition of $x_i$, this is always the case for STARKAD$^\pi$ if $t$ is a power of 2.

of $M^l$. The opposite is not true in general: Given an eigenvalue $\lambda$ of $M^l$, it is possible that $\lambda^{1/l}$ does not exist, which means that there is no corresponding eigenvalue for $M$.

This fact has an impact on the existence of infinitely long subspace trails. Indeed, in the case in which there exists $l \geq 2$ s.t. $M^l$ has more eigenvalues than $M$, it is potentially possible to set up an iterative subspace trail which is not invariant (and for which no S-box is active) for any number of rounds.

**Theorem 2.** *Given a P-SPN scheme with $s$ S-boxes defined as in Eq. (2), let $M \in \mathbb{F}^{t\times t}$ be an invertible matrix. Let $\lambda_1^{(l)}, \ldots, \lambda_\tau^{(l)}$ be the eigenvalues of $M^l$ for some $l \geq 1$, and let $\mathcal{P}_1^{(l)}, \ldots, \mathcal{P}_\tau^{(l)}$ be their corresponding eigenspaces (where $\tau \leq t$). For each $r \geq 1$, let $\mathcal{IS}^{(r)}$ be the subspace defined as*

$$\mathcal{IS}^{(r)} = \left\langle \mathcal{S}^{(r)} \cap \mathcal{P}_1^{(r)}, \mathcal{S}^{(r)} \cap \mathcal{P}_2^{(r)}, \ldots, \mathcal{S}^{(r)} \cap \mathcal{P}_\tau^{(r)} \right\rangle,$$

*where $\mathcal{S}^{(r)}$ is the subspace constructed as in Eq. (3) s.t. no S-box is active in the first $r$ rounds. If $1 \leq \dim\left(\mathcal{IS}^{(r)}\right) < t$, an infinitely long iterative subspace trail of the form*

$$\left\{\mathcal{IS}^{(r)}, M \cdot \mathcal{IS}^{(r)}, M^2 \cdot \mathcal{IS}^{(r)}, \ldots, M^{r-1} \cdot \mathcal{IS}^{(r)}\right\}$$

*is generated.*

*Proof.* The proof of this result is analogous to the ones given before. It is sufficient to note that *(1)* no S-box is active due to the definition of $\mathcal{S}^{(r)}$ (see Eq. (3)) and that *(2)* the subspace trail is iterative with a period equal to $r$ since $\mathcal{IS}^{(r)}$ is constructed using the eigenspaces of $M^r$. □

We point out that this result reduces to the previous one in the case in which $l = 1$, since $\mathcal{S}^{(1)} = \langle e_{s+1}, \ldots, e_t \rangle$, and that it provides only a sufficient condition.

**Low Multiplicative Order.** This result also includes the case in which the matrix has a low multiplicative order, as shown in the following corollary.

**Corollary 1.** *Theorem 2 implies the result presented in Proposition 3.*

*Proof.* Assume there exists $l$ such that $M^l = \mu \cdot I$. Then $e_1, \ldots, e_t$ are all eigenvectors of $M^l$ with eigenvalue $\mu$ (equivalently, the space $\mathbb{F}^t$ is an eigenspace of $M^l$ w.r.t. the eigenvalue $\mu$). Moreover, let $\mathcal{S}^{(l)}$ be the subspace constructed as in Eq. (3) such that no S-box is active in the first $l$ rounds. Since $\langle e_1, \ldots, e_t \rangle$ is an eigenspace of $M^l$ corresponding to the eigenvalue $\mu$, it follows that $\mathcal{S}^{(l)}$ is an invariant subspace of $M^l$. Hence, due to the previous considerations, $\left\{\mathcal{S}^{(l)}, M \cdot \mathcal{S}^{(l)}, M^2 \cdot \mathcal{S}^{(l)}, \ldots, M^{l-1} \cdot \mathcal{S}^{(l)}\right\}$ is an infinitely long iterative (constant-dimensional) subspace trail. □

We remark that the two conditions are not equivalent (namely, Proposition 3 does not imply in general Theorem 2), as shown in the following concrete example.

**Example.** Consider the circulant matrix $M = \text{circ}(a, b, c, d)$ over $\mathbb{F}^4$. Its eigenvalues are

$$a + b + c + d, \quad \pm\sqrt{a^2 + b^2 - 2ac + c^2 - 2bd + d^2}, \quad a - b + c - d,$$

while the eigenvalues of $M^2$ are $(a+b+c+d)^2$, $a^2+b^2-2ac+c^2-2bd+d^2$ and $(a-b+c-d)^2$. Since $x \mapsto x^2$ is not a permutation over $\mathbb{F}_p$ for a prime $p \geq 3$ (see Hermite's criterion), there exist $a, b, c, d$, s.t. $a^2 + b^2 - 2ac + c^2 - 2bd + d^2$ is not a square. Hence, for certain values of $a, b, c, d \in \mathbb{F}_p$, it is possible that $M$ has two eigenvalues, while $M^2$ has always four eigenvalues.[11] This fact can be exploited in order to construct a matrix $M$ that is not a multiple of the identity and for which an infinitely long iterative subspace trail exists. Given a P-SPN scheme over $(\mathbb{F}_p)^5$ with $s = 1$, a concrete example of such a matrix is

$$\mathfrak{M} = \begin{pmatrix} x & y & w & y & w \\ z_0 & a & b & c & d \\ z_1 & b & c & d & a \\ z_2 & c & d & a & b \\ z_3 & d & a & b & c \end{pmatrix}$$

for particular values of $a, b, c, d, x, y, w, z_j \in \mathbb{F}_p$ s.t. *(1)* the matrix is invertible and it provides full diffusion (at word level after a finite number of rounds) for cryptographic purposes and *(2)* the circulant matrix $\text{circ}(a, b, c, d)$ has only 2 eigenvalues.

The iterative (non-invariant) subspace trail is thus given by $\left\{\mathcal{IS} = \langle (0, 0, 1, 0, -1)^T \rangle, \mathfrak{M} \cdot \mathcal{IS} = \langle (0, b-d, c-a, d-b, a-c)^T \rangle \right\}$, where $\mathfrak{M}^2 \cdot \mathcal{IS} = \mathcal{IS}$ and where $\mathfrak{M}^2 \neq \mu \cdot I$ for each $\mu \in \mathbb{F}_p$ (we refer to Appendix B for more details).

## 3.5 Infinitely Long Iterative Subspace Trails with No Active S-Boxes: A Necessary and Sufficient Condition

Consider the case of a Cauchy matrix $M$ generated as in [25] (recalled in Section 3.3) for $t = 24$ and $\mathbb{F}_{2^n}$, where $n = 63$. As shown in [36, Page 20], the subspace $\mathcal{S}^{(5)}$ defined as in Eq. (3) satisfies $M \cdot \mathcal{S}^{(5)} = \mathcal{S}^{(5)}$ and $(M \cdot x)[1] = 0$ for all $x \in \mathcal{S}^{(5)}$.

The reason why we highlight this fact is that it provides an example of a matrix for which our conditions given before are only sufficient but not necessary. In other words, if the previous condition (namely, Theorem 2) is both necessary and sufficient, then the subspace $\mathcal{S}^{(5)}$ must be related to the eigenspaces of $M$. However, by simple practical tests, this is not the case since $M^j$ for $j \in [1, 5]$ does not have any eigenvalues and eigenspaces.

More generally, let $d$ be the dimension of a (generic) invariant subspace $\mathcal{S} = \langle s_1, \ldots, s_d \rangle$ for a $t \times t$ matrix $M$. Such a subspace is related to the eigenvectors of $M$ if there exist $\alpha_1, \ldots, \alpha_d, \mathfrak{A} \in \mathbb{F}$ (with $(\alpha_1, \ldots, \alpha_d) \neq (0, \ldots, 0)$ and $\mathfrak{A} \neq 0$) s.t. $M \cdot (\alpha_1 \cdot s_1 + \alpha_2 \cdot s_2 + \cdots + \alpha_d \cdot s_d) = \mathfrak{A} \cdot (\alpha_1 \cdot s_1 + \alpha_2 \cdot s_2 + \cdots + \alpha_d \cdot s_d)$ (by definition). Since $M \cdot s_i = \sum_j \beta_j^i \cdot s_j$ for certain $\beta_j^i \in \mathbb{F}$ ($\mathcal{S}$ is invariant) and since $\{s_i\}_i$ are linearly independent, a non-trivial solution of the previous equality exists if there is (at least) one $\mathfrak{A} \neq 0$ s.t.

$$\det \begin{pmatrix} \beta_1^1 - \mathfrak{A} & \beta_1^2 & \ldots & \beta_1^d \\ \beta_2^1 & \beta_2^2 - \mathfrak{A} & \ldots & \beta_2^d \\ \vdots & \vdots & \ddots & \vdots \\ \beta_s^1 & \beta_d^2 & \ldots & \beta_d^d - \mathfrak{A} \end{pmatrix} = 0.$$

If this is not the case (remember that this can happen since $\mathbb{F}$ is not algebraically closed), then $\mathcal{S}$ is an invariant subspace but it is not related to the eigenspaces of $M$. Hence, we can deduce the following.

**Theorem 3.** *Given a P-SPN scheme with $s$ S-boxes defined as in Eq. (2), 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. $\mathcal{S}^{(i)} = \left(M \cdot \mathcal{S}^{(i)}\right)$ 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[12] $i \geq l$ s.t. $\mathcal{S}^{(i)} = \left(M^l \cdot \mathcal{S}^{(i)}\right)$ and $\mathcal{IS} \subseteq \mathcal{S}^{(i)}$.*

[11] E.g., given $(a, b, c, d) = (1, 1, 2, 3)$, $a^2 + b^2 - 2ac + c^2 - 2bd + d^2$ is a square in $\mathbb{F}_{11}$, but not in $\mathbb{F}_{13}$.

[12] Note that there cannot exist an iterative subspace trail with no active S-boxes where $i < l$. Indeed, assume e.g. $l = i+1$ and $\mathcal{S}^{(i)} \neq \mathcal{S}^{(i+1)}$. Hence, the subspace trail $\{\mathcal{S}^{(i)}, M \cdot \mathcal{S}^{(i)}, \ldots, M^{i-1} \cdot \mathcal{S}^{(i)}, M^i \cdot \mathcal{S}^{(i)}\}$ should be iterative. However, there is no guarantee that no S-box is active in $M^i \cdot \mathcal{S}^{(i)}$ due to the definition of $\mathcal{S}^{(i)}$. This can happen only in the case $\mathcal{S}^{(i)} = \mathcal{S}^{(i+1)}$, which implies that the period is actually $i + 1$.

*Proof.* Let us focus on the case $l = 1$ (analogous proof for $l \geq 2$). First, we show that $\mathcal{IS} = \mathcal{S}^{(i)} = \left(M \cdot \mathcal{S}^{(i)}\right)$ is an infinitely long invariant subspace trail (with no active S-boxes). This follows immediately by the definition of $\mathcal{S}^{(i)}$ (which implies that no S-box is active) and by Lemma 1 (which, together with $\mathcal{S}^{(i)} = \left(M \cdot \mathcal{S}^{(i)}\right)$, implies that $\mathcal{S}^{(j)} = \mathcal{S}^{(j+1)}$ for each $j \geq i$). Indeed, under the assumption $\mathcal{S}^{(i)} = \left(M \cdot \mathcal{S}^{(i)}\right)$, it follows that $\forall x \in \mathcal{S}^{(i)}$: $M \cdot x \in \mathcal{S}^{(i)} \subseteq \mathcal{S}^{(1)} = \langle e_{s+1}, \ldots, e_t \rangle$. Due to the result of Lemma 1, this implies that if $x \in \mathcal{S}^{(i)} = \left(M \cdot \mathcal{S}^{(i)}\right)$, then $x \in \mathcal{S}^{(i+1)}$.

Vice-versa, here we show that given an infinitely long invariant subspace trail $\mathcal{IS}$ (with no active S-boxes), there must exist $i \geq 1$ s.t. $\mathcal{S}^{(i)} = \left(M \cdot \mathcal{S}^{(i)}\right)$ and $\mathcal{IS} \subseteq \mathcal{S}^{(i)}$. To do this, observe that all pairs of texts which do not activate any S-box in the first $i$ rounds are in the same coset of $\mathcal{S}^{(i)}$. Focusing on the linear layer, note that a subspace $\mathcal{X}$ is invariant if and only if $M \cdot \mathcal{X} = \mathcal{X}$. This means that $\mathcal{IS} \subseteq \mathcal{S}^{(i)}$, where $\mathcal{S}^{(i)}$ must satisfy $\mathcal{S}^{(i)} = M \cdot \mathcal{S}^{(i)}$. The result follows immediately. □

For example, for the Cauchy matrix $M$ generated as in [25] and recalled before, the subspace $\mathcal{S}^{(5)}$ satisfies $M \cdot \mathcal{S}^{(5)} = \mathcal{S}^{(5)}$. Moreover, the following results hold.

**Corollary 2.** *The infinitely long subspace trail with inactive S-boxes presented in Theorem 1 and Theorem 2 satisfies Theorem 3.*

*Proof.* Let us focus here on the case $l = 1$ (analogous proof for $l \geq 2$). Let $\mathcal{P}_1, \ldots, \mathcal{P}_\tau$ be the eigenspaces of $M$, and let $\mathcal{P}'_j = \mathcal{P}_j \cap \mathcal{S}^{(1)}$ for each $j \in \{1, \ldots, \tau\}$. Our goal is to show that there exists $i \geq 1$ s.t. $\mathcal{S}^{(i)} = \left(M \cdot \mathcal{S}^{(i)}\right)$ and $\langle \mathcal{P}'_1, \ldots, \mathcal{P}'_\tau \rangle \subseteq \mathcal{S}^{(i)}$.

Since $\langle \mathcal{P}'_1, \ldots, \mathcal{P}'_\tau \rangle$ is invariant, $\langle \mathcal{P}'_1, \ldots, \mathcal{P}'_\tau \rangle \subseteq \mathcal{S}^{(i)}$ for each $i \geq 1$. Hence, let $\mathcal{S}^{(i)} = \langle \mathcal{P}'_1, \ldots, \mathcal{P}'_\tau, \mathcal{Q}^{(i)} \rangle$ where $\mathcal{Q}^{(i)} \cap \mathcal{P}'_j = \emptyset$ for each $j \in \{1, \ldots, \tau\}$ (where $0 \leq \dim\left(\mathcal{Q}^{(i)}\right) < \dim\left(\mathcal{S}^{(i)}\right)$). By the definition of the subspace $\mathcal{Q}^{(i)}$ and since $\mathcal{S}^{(i+1)} \subseteq \mathcal{S}^{(i)}$, it follows that $\mathcal{Q}^{(i+1)} \subseteq M \cdot \mathcal{Q}^{(i)}$, and therefore either $\mathcal{Q}^{(i+1)} = M \cdot \mathcal{Q}^{(i)}$ or $\mathcal{Q}^{(i+1)} \subset M \cdot \mathcal{Q}^{(i)}$ (namely, $\dim\left(Q^{(i+1)}\right) < \dim\left(M \cdot \mathcal{Q}^{(i)}\right)$). Thus, there must exist $i \geq 1$ s.t. either $\mathcal{Q}^{(j+1)} = M \cdot \mathcal{Q}^{(j)}$ for each $j \geq i$ or $\dim\left(\mathcal{Q}^{(j)}\right) = 0$ for each $j \geq i$. This concludes the proof. □

**Theorem 4.** *Given a P-SPN scheme with $s$ S-boxes defined as in Eq. (2), let $M \in \mathbb{F}^{t \times t}$ be an invertible matrix. If $M$ is diagonalizable, the result regarding the existence of an infinitely long subspace trail with inactive S-boxes of Theorem 1 and Theorem 2 provides both a necessary and a sufficient condition. In other words, under the assumption that $M$ is diagonalizable (where $\mathcal{P}_1^{(l)}, \ldots, \mathcal{P}_\tau^{(l)}$ are the eigenspaces of $M^l$),*

*(1) a subspace $\mathcal{IS}$ generates an infinitely long invariant subspace trail (with no active S-boxes) if and only if $\mathcal{IS} = \langle \mathcal{P}_1 \cap \langle e_{s+1}, \ldots, e_t \rangle, \ldots, \mathcal{P}_\tau \cap \langle e_{s+1}, \ldots, e_t \rangle \rangle$, and*

*(2) a subspace $\mathcal{IS}$ generates an infinitely long iterative subspace trail (with no active S-boxes) of period $l \geq 2$ if and only if $\mathcal{IS} = \left\langle \mathcal{P}_1^{(l)} \cap \mathcal{S}^{(l)}, \ldots, \mathcal{P}_\tau^{(l)} \cap \mathcal{S}^{(l)} \right\rangle$, where $\mathcal{S}^{(l)}$ is the subspace constructed as in Eq. (3) s.t. no S-box is active in the first $l$ rounds.*

*Proof.* Let $\mathcal{P}_1, \ldots, \mathcal{P}_\tau$ be the eigenspaces of the matrix $M$. Since $M$ is diagonalizable, $\dim(\mathcal{P}_1) + \cdots + \dim(\mathcal{P}_\tau) = t$ and $M^l$ is diagonalizable (with the same eigenspace). It follows that a subspace $\mathcal{X}$ satisfies $M^l \cdot \mathcal{X} = \mathcal{X}$ (for $l \geq 1$) if and only if there exists $I := \{\iota_1, \ldots, \iota_{|I|}\} \subseteq \{1, \ldots, \tau\}$ s.t. $\mathcal{X} = \langle \mathcal{P}'_{\iota_1}, \ldots, \mathcal{P}'_{\iota_{|I|}} \rangle$, where $\mathcal{P}'_j$ is a non-null subspace of $\mathcal{P}_j$. The result follows immediately by combining this fact with the results of Theorem 3. □

**An Open Problem.** Before going on, we mention that a possible future open problem is to find a direct relation between the result of Theorem 3 and the properties of the matrix $M$ (similar to the sufficient conditions on $M$ for the existence of an infinitely long subspace trail given in Theorem 1 and Theorem 2). To achieve this result, one idea could be to work over the algebraic closure[13] $\mathbb{F}^\star$ of the field $\mathbb{F}$. Indeed, a field $\mathbb{F}$ is algebraically closed if and only if for each natural number $t$ every linear map over $\mathbb{F}^t$ has some eigenvectors.[14]

### 3.6 About Infinitely Long Iterative Subspace Trail with Inactive S-Boxes

As finally question, one may ask if there exists an example of a P-SPN scheme for which there exists no invariant subspace trail and at the same time there exists an iterative subspace trail with inactive S-boxes. As we are going to show this is not possible.

**Proposition 5.** *Consider a P-SPN scheme with $s$ S-boxes defined as in Eq. (2). The existence of an iterative subspace trail with inactive S-boxes is only possible in the case in which there exists an invariant subspace trail with inactive S-boxes.*

*Proof.* As shown in Proposition 1, let $\{\mathcal{V}_1, \ldots, \mathcal{V}_r\}$ be an infinitely long iterative subspace trail of period $r$ (with inactive S-boxes). If $\dim(\langle \mathcal{V}_1, \ldots, \mathcal{V}_r \rangle) < t$, then $\langle \mathcal{V}_1, \ldots, \mathcal{V}_r \rangle$ generates an infinitely long invariant subspace trail. Hence, if $\dim(\langle \mathcal{V}_1, \ldots, \mathcal{V}_r \rangle) = t$, it would be possible that an iterative subspace trail with inactive S-boxes exists and at the same time no invariant subspace trail exists. However, note that $\dim(\langle \mathcal{V}_1, \ldots, \mathcal{V}_r \rangle) = t$ can *never* occur in the case of inactive S-boxes. Indeed, since the first $s$ words of $\mathcal{V}_1, \ldots, \mathcal{V}_r$ are equal to zero (in order to guarantee that no S-box is active), it is not possible that $\langle \mathcal{V}_1, \ldots, \mathcal{V}_r \rangle$ generates the full space $\mathbb{F}^t$. □

This does not result in iterative subspace trails with inactive S-boxes being useless. Indeed, let $\{\mathcal{V}_1, \ldots, \mathcal{V}_r\}$ be an infinitely long iterative subspace trail of period $r$ (with inactive S-boxes). If $\dim(\mathcal{V}_1) < \dim(\langle \mathcal{V}_1, \ldots, \mathcal{V}_r \rangle)$ (note: *strictly* less), then the data cost to set up the iterative subspace trail may be much less than the cost to set up an invariant subspace trail. This can be crucial in scenarios in which there is a limitation on the data allowed for an attack.

# 4 Practical Tests (Inactive S-Boxes)

In this section, we first present an algorithm which can be used to find vulnerabilities and to detect "weak" matrices (w.r.t. the attacks presented before). Secondly, we test several matrices over $\mathbb{F}_p$ and over $\mathbb{F}_{2^n}$ to give an idea of the percentage of "weak" matrices.

### 4.1 Algorithm for Detecting "Weak" Matrices

Algorithm 1 is based on the results just presented in the previous section. It is designed in order to distinguish the case in which the infinitely long subspace trail is related to the eigenspaces of $M^l$ for $l \geq 1$ or not.

Here we focus on the condition $1 \leq i \leq \left\lfloor \frac{t-s}{s} \right\rfloor$, and we explain why it is sufficient to detect all infinitely long subspace trails without active S-boxes. Since $\mathcal{S}^{(i+1)} = \mathcal{S}^{(i)} \cap (M \cdot \mathcal{S}^{(i)})$ and $\mathcal{S}^{(i+1)} \subseteq \mathcal{S}^{(i)}$, only two cases are possible:

*(1)* $\mathcal{S}^{(i+1)} = \mathcal{S}^{(i)}$: In this case, an invariant subspace trail (with inactive S-boxes) generated by $\mathcal{S}^{(i+1)} = M \cdot \mathcal{S}^{(i)} = \mathcal{S}^{(i)}$ exists.

*(2)* $\mathcal{S}^{(i+1)} \subset \mathcal{S}^{(i)}$, where $\dim\left(\mathcal{S}^{(i+1)}\right) = \dim\left(\mathcal{S}^{(i)}\right) - 1$: In this case, an iterative subspace trail (with inactive S-boxes) $\mathcal{S}^{(i)} = M^l \cdot \mathcal{S}^{(i)}$ for $2 \leq l \leq i$ *may* exist.

[13] A field $\mathbb{F}$ is *algebraically closed* if every nonconstant polynomial in $\mathbb{F}[X]$ (the univariate polynomial ring with coefficients in $\mathbb{F}$) has a root in $\mathbb{F}$. For example, no finite field $\mathbb{F}$ is algebraically closed, because if $a_1, a_2, \ldots, a_n$ are all the elements of $\mathbb{F}$, then the polynomial $(x - a_1)(x - a_2) \cdots (x - a_n) + 1$ has no root in $\mathbb{F}$. By contrast, the field of complex numbers is algebraically closed.

[14] A linear map over a field $\mathbb{F}$ has an eigenvector if and only if its characteristic polynomial has a root. Therefore, when $\mathbb{F}$ is algebraically closed, every linear map of $\mathbb{F}^n$ has some eigenvector.

**Algorithm 1:** Determining the existence of (iterative) infinitely long subspace trails without active S-boxes, using Theorem 1, Theorem 2, and Theorem 3.

```
   Data: P-SPN scheme over $\mathbb{F}^t$ with $s$ S-boxes applied to the first $s$ words (where
         the S-box has no linear structure).
   Result: 1 if an (iterative) infinitely long subspace trail exists, 0 otherwise.
1  $flag_1 \leftarrow 0$, $flag_2 \leftarrow 0$.
2  $T \leftarrow \emptyset$. // $T$ stores all iterative subspace trais found
3  Let $\mathcal{S}^{(i)}$ denote the subspace s.t. no S-box is active in the first $i$ rounds
     (Definition 5), and let $\left\{\mathcal{P}_1^{(i)}, \mathcal{P}_2^{(i)}, \ldots, \mathcal{P}_\tau^{(i)}\right\}$ denote the eigenspaces[15]of $M^i$.
4  for $i \leftarrow 1$ to $\left\lfloor \frac{t-s}{s} \right\rfloor$ do
5      if $\exists \mu \in \mathbb{F}$ s.t. $M^i = \mu \cdot I$ (where $I$ is the identity matrix) then
6          return 1: Discard the matrix $M$ (due to low multiplicative order).
7      $\mathcal{IS}^{(i)} \leftarrow \left\langle \mathcal{P}_1^{(i)} \cap \mathcal{S}^{(i)}, \mathcal{P}_2^{(i)} \cap \mathcal{S}^{(i)}, \ldots, \mathcal{P}_\tau^{(i)} \cap \mathcal{S}^{(i)} \right\rangle$.
8      if $\dim\left(\mathcal{IS}^{(i)}\right) \geq 1$ and $\mathcal{IS}^{(i)} \neq \mathbb{F}^t$ then
9          return 1: Discard the matrix $M$ (due to eigenspaces of $M^i$).
10     if $\dim\left(\mathcal{S}^{(i)}\right) \geq 1$ then
11         if $\mathcal{S}^{(i)} = \left(M \cdot \mathcal{S}^{(i)}\right)$ then
12             $flag_1 \leftarrow 1$;
13             $T \leftarrow T \cup \mathcal{S}^{(i)}$;
14         for $j \leftarrow 2$ to $i$ do
15             if $\mathcal{S}^{(i)} = \left(M^j \cdot \mathcal{S}^{(i)}\right)$ then
16                 $flag_2 \leftarrow 1$;
17                 $T \leftarrow T \cup \mathcal{S}^{(i)}$;
18 if $flag_2 = 1$ then
19     return 1: Discard the matrix $M$ (due to existence of invariant and iterative
         subspace trails $T$ – see Theorem 3).
20 else if $flag_1 = 1$ then
21     return 1: Discard the matrix $M$ (due to existence of invariant subspace trail(s)
         $T$ – see Theorem 3).
22 else
23     return 0: No (iterative) infinitely long subspace trail found.
```

Moreover, remember that if $\mathcal{S}^{(i+1)} \neq \mathcal{S}^{(i)}$, then $\dim\left(\mathcal{S}^{\left(\left\lfloor \frac{t-s}{s} \right\rfloor + 1\right)}\right) \leq s-1$, where $i \leq \left\lfloor \frac{t-s}{s} \right\rfloor$.

In other words, since $\mathcal{S}^{(\cdot)}$ are subspaces, the case $\mathcal{S}^{(i+1)} \neq \mathcal{S}^{(i)}$ where $\dim\left(\mathcal{S}^{(i+1)}\right) = \dim\left(\mathcal{S}^{(i)}\right)$ can never happen.[16] As a result, if $\mathcal{S}^{(i+1)} \neq \mathcal{S}^{(i)}$ for each $i \leq \left\lfloor \frac{t-s}{s} \right\rfloor$ and *since Theorem 3 provides both a necessary and a sufficient condition*, $\left\lfloor \frac{t-s}{s} \right\rfloor$ rounds are sufficient to determine if an infinitely long subspace trail with inactive S-boxes exists.

*Remark.* Before going on, we highlight that vulnerable matrices found in Line 5 and Line 8 of Algorithm 1 are also vulnerable to the condition evaluated in Line 10 (as expected, due to the previous results). In other words, it is not necessary to check the condition on the eigenspaces of $M^l$: we decided to do that in order to better understand the percentage of weak matrices whose vulnerability is related to their eigenspaces. Indeed, this percentage

[15] Note that if $\mathcal{P}^{(i)}$ is an eigenspace of $M^i$, it is also an eigenspace of $M^j$ for $j \geq i$.

[16] Assume by contradiction that $\mathcal{S}^{(i+1)} \neq \mathcal{S}^{(i)}$ and $\dim\left(\mathcal{S}^{(i+1)}\right) = \dim\left(\mathcal{S}^{(i)}\right)$. Hence, $\mathcal{S}^{(i+1)} = \langle \mathcal{S}^{(i)} \cap \mathcal{S}^{(i+1)}, \mathcal{X} \rangle$ for a certain subspace $\mathcal{X}$ such that $\dim(\mathcal{X}) \geq 1$. As a result, $\mathcal{S}^{(i+1)}$ is not a subspace of $\mathcal{S}^{(i)}$, which cannot happen (see e.g. Lemma 1).

is high (higher than 95% by our tests).

Moreover, in the part starting in Line 10, it would be sufficient to consider the case $j = 1$ (corresponding to the case of invariant subspace trail), due to the result of Proposition 5. We decided to include $j \geq 2$ for completeness, in the case in which one is interested to look for iterative subspace trails (due to, e.g., a restriction on the data cost, as explained before).[17]

**Computational Cost of Algorithm 1.** First, the computation of the subspace $\mathcal{S}^{(i)}$ requires the resolution of a system of $s$ linear equations, for a total cost of $\mathcal{O}(s^{\omega}) \subseteq \mathcal{O}(t^3)$ (where $2 < \omega \leq 3$). We further regard the complexity of a multiplication of two $t \times t$ matrices as an element of $\mathcal{O}(t^3)$.

In a reduced form, only the condition in Line 10 of Algorithm 1 needs to be evaluated, since it already includes the other conditions. This line is iterated $\left\lfloor \frac{t-s}{s} \right\rfloor$ times and at most $t$ multiplications of $t \times t$ matrices take place in this line. Hence, the total complexity is an element of $\mathcal{O}(t^4)$.

When including all calculations, the computations of the eigenspaces of $M^i$ and more matrix multiplications in general are needed. The eigendecomposition of a $t \times t$ matrix needs a number of field operations in $\mathcal{O}(t^3)$. The total runtime is then also an element in $\mathcal{O}\left(\left\lfloor \frac{t-s}{s} \right\rfloor \cdot t^3\right) \subseteq \mathcal{O}(t^4)$, however, the hidden constants are significantly larger. In practical tests, evaluating only the condition in Line 10 thus leads to a considerably better performance.

**Implementation.** We make our implementation available online[18]. This tool can be used to detect vulnerabilities of given matrices over prime field or binary fields.

**Computational Cost in Practice.** There are various ways to implement Algorithm 1 in practice. We decided to store the powers of the input matrix $M$ beforehand, i.e., we compute and store $M, M^2, \ldots, M^l$, where $l$ is the number of iterations. Hence, the memory cost depends on $l$ and is then essentially in $\mathcal{O}(l \cdot t^2)$ for a $t \times t$ matrix $M$.

The runtime is dominated by finding a solution to the system of equations and by building the eigendecomposition of a matrix. Both complexities are in $\mathcal{O}(t^3)$ for $t \times t$ matrices. Just to give some concrete practical numbers, for $n = 16$, the test for a single matrix takes about 30 milliseconds for $t = 4$, while it takes about 600 milliseconds for $t = 12$ (using an Intel Xeon E5-2699v4 with a maximum clock frequency of 3.60 GHz). When omitting the computation of eigenspaces and evaluating only the condition in Line 10 of Algorithm 1 (which is sufficient from the designer's point of view), the test takes about 4 and 50 milliseconds, respectively, for the two cases.

## 4.2 Percentage of "Weak" Linear Layers

We implemented Algorithm 1 in `Sage` and used it to get an idea of the percentage of matrices that are vulnerable to the attack without active S-boxes presented in Section 3.

**Different Classes of Matrices.** For concrete use cases, we set $s = 1$ and we focus on two scenarios, namely random invertible matrices and random Cauchy matrices[19]. As the source for randomness we use `Sage`'s random engine in both cases (and for choosing e.g. the prime numbers). In the first scenario, we create a matrix space, sample random matrices, and finally determine if they are invertible. In the second scenario, we generate

[17] Note that in this case, the pseudo code has to be modified slightly.

[18] https://extgit.iaik.tugraz.at/krypto/linear-layer-tool

[19] We recall that $M \in \mathbb{F}^{t \times t}$ is a Cauchy matrix if there exists $\{x_i, y_i\}_{i=1}^{t}$ s.t. $M_{i,j} = \frac{1}{x_i + y_j}$, where for each $i \neq j : x_i \neq x_j$, $y_i \neq y_j$, $x_i + y_j \neq 0$. Cauchy matrices are MDS matrices.

**Table 1:** Percentage of vulnerable matrices for Algorithm 1 and orders $t$, when considering prime fields $\mathrm{GF}(p)$.

| *Random Invertible* | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| $\lceil \log_2(p) \rceil$ | 4 | 8 | 4 | 8 | 8 | 16 | 8 | 16 |
| $t$ | 3 | 3 | 4 | 4 | 8 | 8 | 12 | 12 |
| Vulnerable (%) | 7.42 | 0.54 | 7.46 | 0.46 | 0.42 | $< 0.01$ | 0.38 | $< 0.01$ |
| *MDS (Random Cauchy)* | | | | | | | | |
| $\lceil \log_2(p) \rceil$ | 4 | 8 | 4 | 8 | 8 | 16 | 8 | 16 |
| $t$ | 3 | 3 | 4 | 4 | 8 | 8 | 12 | 12 |
| Vulnerable (%) | 7.72 | 0.50 | 5.76 | 0.58 | 0.50 | 0.02 | 0.38 | $< 0.01$ |

**Table 2:** Percentage of vulnerable matrices for Algorithm 1 and orders $t$, when considering binary fields $\mathrm{GF}(2^n)$.

| *Random Invertible* | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| $n$ | 4 | 8 | 4 | 8 | 8 | 16 | 8 | 16 |
| $t$ | 3 | 3 | 4 | 4 | 8 | 8 | 12 | 12 |
| Vulnerable (%) | 6.32 | 0.60 | 5.66 | 0.38 | 0.34 | $< 0.01$ | 0.40 | $< 0.01$ |
| *MDS (Random Cauchy)* | | | | | | | | |
| $n$ | 4 | 8 | 4 | 8 | 8 | 16 | 8 | 16 |
| $t$ | 3 | 3 | 4 | 4 | 8 | 8 | 12 | 12 |
| Vulnerable (%) | 6.60 | 0.42 | 5.38 | 0.40 | 0.66 | $< 0.01$ | 0.38 | $< 0.01$ |

Cauchy matrices using random (and valid) starting sequences. We tested all matrices using both prime fields and binary fields, focusing on square matrices of order $t \in \{3, 4, 8, 12, 16\}$ and on fields with a size of $n \in \{4, 8, 16\}$ (and $\lceil \log_2(p) \rceil \in \{4, 8, 16\}$ for prime fields). Moreover, we tested our algorithm on the concrete matrices used to instantiate Starkad and Poseidon. We present these results in Appendix D.1.

**Concrete Results.** The sample size for all tests was set to 50000. While a matrix chosen completely at random (or without considering our results) may be vulnerable with a significant probability, it is easy to choose a matrix which is not vulnerable to the attacks presented above. Namely, given the estimated percentages of vulnerable matrices found in the tables above, the probability of finding a "secure" matrix (w.r.t. our results) is already quite high after trying two or more different matrices. In other words, our tool can easily be used to find matrices which are not vulnerable to the attacks presented in Section 3.

Regarding the tables, we can immediately see that the choice of $p$ (or $n$) has an impact on the number of vulnerable matrices. Specifically, increasing $p$ (or $n$) tends to result in a higher probability for a matrix to be secure against the attacks presented here.

Finally, we briefly mention that a very high percentage of "weak" matrices (higher than 95% in practical tests) are identified due to their eigenspaces, that is due to the results from Theorem 1 - Theorem 2 rather than by the generic results from Theorem 3.

# 5 Subspace Trails for P-SPN Schemes with Active S-Boxes

Until now, we focused on the case in which no S-box is active. Here, we analyze the scenario in which S-boxes are active.

## 5.1 Preliminaries: Subspace Trails and Truncated Differentials

We first present a generic result regarding the minimum number of rounds for which it is possible to set up a subspace trail with a probability of 1.

**Proposition 6.** *Given a partial SPN scheme over $\mathbb{F}^t$ with $s < \lceil t/2 \rceil$ S-boxes defined as in Eq. (2), there exists a subspace trail with prob. 1 on at least $2 \cdot \left\lfloor \frac{t-s}{s} \right\rfloor$ rounds, defined by*

$$\left\{ \mathcal{S}^{(\lfloor \frac{t-s}{s} \rfloor)}, M \cdot \mathcal{S}^{(\lfloor \frac{t-s}{s} \rfloor)}, \ldots, M^{\lfloor \frac{t-s}{s} \rfloor - 1} \cdot \mathcal{S}^{(\lfloor \frac{t-s}{s} \rfloor)}, \mathcal{A}^{(1)}, \ldots, \mathcal{A}^{(\lfloor \frac{t-s}{s} \rfloor)} \right\},$$

*where $\mathcal{S}^{(i)}$ is defined as in Eq. (3) and where $\mathcal{A}^{(i)} := \left\langle M(e_1), \ldots, M(e_s), M \cdot \mathcal{A}^{(i-1)} \right\rangle$ for $i \geq 1$ (where $\mathcal{A}^{(0)} := M^{\lfloor \frac{t-s}{s} \rfloor - 1} \cdot \mathcal{S}^{(\lfloor \frac{t-s}{s} \rfloor)}$).*

As for Proposition 2, this well-known result (whose proof can be found in Appendix C) only depends on the number of S-boxes, and no assumption on the matrix $M$ is made. Similar to the case presented in Section 3.1, note that depending on the details of the linear layer, a longer subspace trail of dimension 1 can be set up.

## 5.2 Infinitely Long Subspace Trail with Active S-Boxes: A Sufficient Condition on the Linear Layer $M$

Next, we analyze infinitely long subspace trails in the case of active S-boxes. Working as in Section 3, here we study which properties a linear layer must satisfy in order to set up an infinitely long subspace trail also in the case of active S-boxes.

### 5.2.1 Infinitely Long Invariant Subspace Trails with Active S-Boxes

Using the approach proposed in Section 3.2, we first focus on the case of invariant subspace trails with active S-boxes.

**Theorem 5.** *Given a P-SPN scheme with $s$ S-boxes defined as in Eq. (2), let $M \in \mathbb{F}^{t \times t}$ be an invertible matrix. Let $\lambda_1, \ldots, \lambda_\tau$ be the eigenvalues of $M$, and let $\mathcal{P}_1, \ldots, \mathcal{P}_\tau$ be the corresponding eigenspaces (where $\tau \leq t$). Let $I = \{i_1, \ldots, i_{|I|}\} \subseteq \{1, \ldots, s\}$ be the indices of the words with active S-boxes (where $I \neq \emptyset$), and let*

$$\mathcal{IS} = \langle \mathcal{P}'_1, \ldots, \mathcal{P}'_\tau \rangle ,$$

*where $\mathcal{P}'_h$ is a subspace of $\mathcal{P}_h$ for each $h \in \{1, \ldots, \tau\}$. If $1 \leq \dim(\mathcal{IS}) < t$ and if $\mathcal{IS}$ satisfies*

1. *$\mathcal{IS} \cap \left\langle e_{i_1}, \ldots, e_{i_{|I|}}, e_{s+1}, \ldots, e_t \right\rangle = \mathcal{IS}$, and*
2. *$\forall i \in I \subseteq \{1, \ldots, s\} : \quad \mathcal{IS} \cap \langle e_i \rangle = \langle e_i \rangle$,*

*then $\mathcal{IS}$ generates an infinitely long invariant subspace trail with active S-boxes.*

*Proof.* The first condition ensures that no $l$-th word is active, where $l \notin I$. For each $i$-th active word, where $i \in I$, the second condition implies that the entire space $\langle e_i \rangle$ is included in $\mathcal{IS}$. The consequence is that, when applying the S-box, the subspace remains the same.

As for the results given in the previous sections, this subspace remains invariant through the linear layer since it is defined using the eigenspaces of $M$. Hence, $\mathcal{IS}$ results in an infinitely long invariant subspace trail. □

Note that the number of active S-boxes in the previous subspace trail is proportional to the number of rounds (so, potentially "infinite"). As before, we emphasize that, in general, the previous observation provides only a sufficient condition.

The result just given is hard to exploit in practice, since a direct construction of $\mathcal{IS}$ is missing. This means that one has to consider all possible subspaces of the eigenspaces $\mathcal{P}$, which is more expensive when working over a large field $\mathbb{F}$ (e.g., $\mathbb{F}_p$ or $\mathbb{F}_{2^n}$ for large $p$ or $n$) in the case in which one of these eigenspaces has a dimension larger than 1. Indeed, since the number of subspaces of $\mathcal{X} \subseteq \mathbb{F}^t$ of dimension $\dim(\mathcal{X}) - 1 \geq 1$ is an element of $\mathcal{O}(|\mathbb{F}|^{\dim(\mathcal{X})-1})$ (see e.g. [33] for details), the cost of this step *could* be proportional to the size of the field $\mathbb{F}$. In the following, we show how to solve this problem.

**Example.** Given a P-SPN scheme with $s = 1$, consider the following $4 \times 4$ matrix $M$ defined over $\mathbb{F}$:

$$\mathfrak{M} = \begin{pmatrix} 0 & (1 - M_{1,3} \cdot b - M_{1,4} \cdot c)/a & M_{1,3} & M_{1,4} \\ a & (-M_{2,3} \cdot b - M_{2,4} \cdot c)/a & M_{2,3} & M_{2,4} \\ b & (-M_{3,3} \cdot b - M_{3,4} \cdot c)/a & M_{3,3} & M_{3,4} \\ c & (-M_{4,3} \cdot b - M_{4,4} \cdot c)/a & M_{4,3} & M_{4,4} \end{pmatrix}, \tag{4}$$

where $a \neq 0$. A proper choice of $a, b, c$ and $\mathfrak{M}_{\cdot,\cdot}$ provides invertibility and "full diffusion" (at word level after a finite number of rounds) for cryptographic purposes. The subspace

$$\mathcal{IS} = \left\langle e_1 = (1,0,0,0)^T, v = (0,a,b,c)^T \right\rangle,$$

where $\mathfrak{M} \cdot e_1 = v$ and $\mathfrak{M} \cdot v = e_1$, is invariant under the round transformation for any number of rounds. Indeed, since the first word can take every value and because the S-box is applied only to this word, $\mathcal{IS}$ remains invariant (note that the S-box is active). Hence, this is a concrete example of an infinitely long invariant subspace trail with active S-boxes, where $\mathcal{P}_1 = \langle v + e_1 \rangle$ and $\mathcal{P}_2 = \langle v - e_1 \rangle$ are the eigenspaces of the matrix $M$ that satisfy the conditions given in the previous theorem (we refer to Appendix E.1 for other examples).

Lastly, we remark that matrices of the form Eq. (4) are currently used in the literature: For example, the circulant almost-MDS matrix over $\mathbb{F}_{2^n}$ defined as $\text{circ}(0, 1, 1, 1)$ is used in Midori [9] and QARMA [8].

**A P-SPN Scheme Vulnerable to Invariant Subspace Trails with Active S-Boxes, but not to Trails with Inactive S-Boxes.** Here we provide an example of a P-SPN scheme vulnerable to subspace trails with active S-boxes, but not to trails with inactive S-boxes. In order to do this, we first propose the following observation for the case $s = 1$.

**Lemma 2.** *Consider a P-SPN scheme with $s = 1$ S-box per round defined as in Eq. (2). Assume there exists a (non-trivial) infinitely long invariant subspace trail $\mathcal{IS}$ with active S-boxes (hence, $1 \leq \dim(\mathcal{IS}) < t$). Given $\mathcal{V}^{(0)} := \mathcal{IS}^c$, let $\mathcal{V}^{(i+1)} = \mathcal{V}^{(i)} \cap (M \cdot \mathcal{V}^{(i)})$ for each $i \geq 0$. An infinitely long (invariant/iterative) subspace trail $\mathcal{IS}' \subseteq \mathcal{IS}^c$ with no active S-boxes exists if and only if there exists an integer $i$ and $0 \leq l \leq i$ s.t. $\mathcal{IS}' \subseteq \mathcal{V}^{(i)}$ and $\mathcal{V}^{(i)} = M^l \cdot \mathcal{V}^{(i)}$, where $M^0 = I$ is the identity matrix.*

The proof of this proposition is analogous to the one presented for Theorem 3. It is sufficient to note that $\langle e_1 \rangle \subseteq \mathcal{IS}$. This implies that $\mathcal{IS}^c \subseteq \langle e_2, \ldots, e_t \rangle$ (that is, $x[1] = 0$ for each $x \in \mathcal{IS}^c$), hence $\mathcal{IS}^c \subseteq \mathcal{S}^{(1)}$ (where $\mathcal{S}^{(1)}$ is defined as in Eq. (3)).

Note that since $\dim(\mathcal{IS}^c) \leq t - 1$, there must exist a finite $j$ s.t. either $\mathcal{V}^{(j+1)} = \mathcal{V}^{(j)}$ or $\dim(\mathcal{V}^{(j)}) = 0$ (see the argument given in Section 4).

Given a P-SPN scheme with $s = 1$, an example for a matrix fulfilling these properties (i.e., leading to a scheme which is not vulnerable in the case of inactive S-boxes, but which allows for invariant subspace trails in the case of active S-boxes) is given by the $4 \times 4$ MDS matrix

$$\mathfrak{M} = \begin{pmatrix} 3 & 1 & 1 & 2 \\ 3 & 4 & 2 & 1 \\ 2 & 1 & 3 & 4 \\ 4 & 1 & 4 & 1 \end{pmatrix}$$

over $\mathbb{F}_p$, where $p = 4\,145\,377\,273$ and $\lceil \log_2(p') \rceil = 32$.

### 5.2.2 Infinitely Long Iterative Subspace Trails with Active S-Boxes

The previous results can be generalized to iterative (non-invariant) subspace trails with active S-boxes by considering the eigenspaces of $M^l$ for $l \geq 2$.

**Theorem 6.** *Given a P-SPN scheme with $s$ S-boxes defined as in Eq. (2), let $M \in \mathbb{F}^{t \times t}$ be an invertible matrix. Let $\lambda_1^{(l)}, \dots, \lambda_\tau^{(l)}$ be the eigenvalues of $M^l$ for a certain $l \geq 1$, and let $\mathcal{P}_1^{(l)}, \dots, \mathcal{P}_\tau^{(l)}$ be the corresponding eigenspaces. For $0 \leq j \leq l-1$, let $I_j = \{i_1^{(j)}, \dots, i_{|I_j|}^{(j)}\} \subseteq \{1, \dots, s\}$ be the positions of the active S-boxes[20] in the $j$-th round.[21] Let $\mathcal{IS} = \langle \mathcal{P}'_1, \dots, \mathcal{P}'_\tau \rangle$, where $\mathcal{P}'_h$ is a certain subspace of $\mathcal{P}_h^{(l)}$ for $h \in \{1, \dots, \tau\}$. If $1 \leq \dim(\mathcal{IS}^{(l)}) < t$ and if $\mathcal{IS}^{(l)}$ satisfies*

1. *$\forall j \in \{0, \dots, l-1\}: \quad (M^j \cdot \mathcal{IS}) \cap \left\langle e_{i_1^{(j)}}, \dots, e_{i_{|I_j|}^{(j)}}, e_{s+1}, \dots, e_t \right\rangle = (M^j \cdot \mathcal{IS})$ and*
2. *$\forall j \in \{0, \dots, l-1\}, \forall i \in I_j \subseteq \{1, \dots, s\}: \quad (M^j \cdot \mathcal{IS}^{(l)}) \cap \langle e_i \rangle = \langle e_i \rangle$,*

*then $\{\mathcal{IS}^{(l)}, M \cdot \mathcal{IS}^{(l)}, \dots, M^{l-1} \cdot \mathcal{IS}^{(l)}\}$ generates an infinitely long iterative subspace trail with active S-boxes.*

*Proof.* The proof of this result is analogous the ones given before. In more detail, the subspace trail is iterative with a period equal to $r$ since $\mathcal{IS}^{(l)}$ is constructed using the eigenspaces of $M^l$ (as in the case of Theorem 2). Secondly, as in the case of Theorem 5, in each round the first condition ensures that no $l$-th word is active, where $l \notin |I_j|$, while the second condition ensures that the entire space $\langle e_i \rangle$ is included in $\mathcal{IS}$ for each $i$-th active word (where $i \in |I_j|$). The consequence is that, when applying the S-box, the subspace remains the same. The result follows immediately. □

**Examples.** Given a P-SPN scheme with $s = 1$, consider again the $4 \times 4$ matrix $\mathfrak{M}$ defined in Eq. (4). The subspace $\mathcal{IS} = \left\langle e_1 = (1,0,0,0)^T \right\rangle$ generates an infinitely long iterative subspace trail with active S-boxes (of period 2) of the form

$$\left\{ \mathcal{IS} = \left\langle e_1 = (1,0,0,0)^T \right\rangle, M \cdot \mathcal{IS} = \left\langle (0,a,b,c)^T \right\rangle \right\},$$

where $I_{2i} = \{1\}$ and $I_{1+2i} = \emptyset$ for each $i \geq 0$.

For a second example, consider the case of a P-SPN scheme over $(\mathbb{F}_{2^n})^4$ with $s = 1$ and $\mathfrak{M} = \operatorname{circ}(0,1,1,1)$. Clearly, both $\left\langle (0,1,1,0)^T \right\rangle$ and $\left\langle (0,1,0,1)^T \right\rangle$ are invariant subspace trails with inactive S-boxes. As shown before, $\left\langle (1,0,0,0)^T, (0,1,1,1)^T \right\rangle$ is an invariant subspace trail with active S-boxes, while $\left\langle (1,0,0,0)^T \right\rangle$ is an iterative (non-invariant) subspace trail with active S-boxes. By combining them, it is possible to set up new iterative subspace trails with active S-boxes, e.g. $\mathcal{IS} = \left\langle (1,0,0,0)^T, (0,1,1,0)^T, (0,1,0,1)^T \right\rangle$. A generalization of this result is presented in Appendix E.2.

[20] Note that in general the number of active S-boxes and their positions do not need to be fixed (if this is the case, it is sufficient to impose $I_x = I_y$ for $x, y < l$).

[21] Note that $I_j = \emptyset$ is also possible. That is, we do not require $|I_j| \geq 1$.

## 5.3 Infinitely Long Iterative Subspace Trails with Active S-Boxes: A Necessary and Sufficient Condition

As done for trails with no active S-boxes, we present a necessary and sufficient condition regarding the existence of infinitely long subspace trails with active S-boxes.

**Theorem 7.** *Given a P-SPN scheme with $s$ S-boxes defined as in Eq. (2), let $M \in \mathbb{F}^{t\times t}$ be an invertible matrix. Given an integer $l \geq 1$, for each $0 \leq j \leq l-1$, let $I_j = \{i_1^{(j)}, \ldots, i_{|I_j|}^{(j)}\} \subseteq \{1, \ldots, s\}$ be the positions of the active S-boxes in the $j$-th round.*[22] *A subspace $\mathcal{IS}$ of $1 \leq \dim(\mathcal{IS}) < t$ generates an infinitely long subspace trail of period $l$ (with active S-boxes if $\exists I_j$ s.t. $|I_j| \geq 1$) if and only if the following conditions are satisfied:*

1. *$\forall j \in \{0, \ldots, l-1\}$ :* $\quad (M^j \cdot \mathcal{IS}) \cap \langle e_{i_1^{(j)}}, \ldots, e_{i_{|I_j|}^{(j)}}, e_{s+1}, \ldots, e_t\rangle = (M^j \cdot \mathcal{IS})$,
2. *$\forall j \in \{0, \ldots, l-1\}, \forall i \in I_j \subseteq \{1, \ldots, s\}$ :* $\quad (M^j \cdot \mathcal{IS}) \cap \langle e_i\rangle = \langle e_i\rangle$,
3. $\mathcal{IS} = (M^l \cdot \mathcal{IS})$.

*If $l = 1$, the subspace trail is invariant.*

*Proof.* As before, the proof of this theorem is analogous to the ones given for Theorem 3 and Theorem 5. We start by showing that a subspace that satisfies the three conditions given before generates an infinitely long iterative subspace trail with active S-boxes. The proof is almost equal to the one given for Theorem 6: The only difference is that the condition that $\mathcal{IS}$ is related to the eigenspace of $M^l$ is replaced by the more generic assumption that $\mathcal{IS}$ satisfies $\mathcal{IS} = (M^l \cdot \mathcal{IS})$.

Vice-versa, assume that a subspace $\mathcal{IS}$ generates an infinitely long iterative subspace trail of period $l$ with active S-boxes. First of all, this can happen if and only if it satisfies the condition $\mathcal{IS} = (M^l \cdot \mathcal{IS})$. The other two conditions are related to the assumption that the S-box does not have any linear structure. Indeed, under this assumption, only two scenarios can happen: The input (and so the output) of the S-box is constant, or the input (and so the output) of the S-box is active (namely, it can take any possible value). The first and the second condition guarantee these two facts in each round. If this would not be the case (namely, if the input of the S-box is neither active nor constant), note that every information about the trail would be lost (due to the assumption on the S-box), and the trail cannot be set up. □

As expected, the results presented in Theorem 5 and Theorem 6 satisfy the previous theorem. This is simply due to the fact that the subspace $\mathcal{IS}$ defined in Theorem 5 and Theorem 6 is related to the eigenspaces of $M^l$, which satisfy the condition $\mathcal{IS} = (M^l \cdot \mathcal{IS})$. We formulate the following corollary.

**Corollary 3.** *The infinitely long subspace trails with active S-boxes presented in Theorem 5 and Theorem 6 satisfy Theorem 7.*

Finally, the following result holds in the case in which $M$ is diagonalizable.

**Theorem 8.** *Given a P-SPN scheme with $s$ S-boxes defined as in Eq. (2), let $M \in \mathbb{F}^{t\times t}$ be an invertible matrix. If $M$ is diagonalizable, the result regarding the existence of an infinitely long invariant subspace trail with active S-boxes of Theorem 5 and Theorem 6 provides both a necessary and a sufficient condition.*

*Proof.* The proof is completely equivalent to the one given before for the case of Theorem 4. In particular, it is sufficient to note that if $M$ is diagonalizable, then a subspace $\mathcal{X}$ satisfies $\mathcal{X} = M^l \cdot \mathcal{X}$ if and only if $\mathcal{X}$ is related to the eigenspaces of $M^l$. □

[22] Note that $I_j = \emptyset$ is also possible. That is, we do not require $|I_j| \geq 1$.

**About Iterative Subspace Trails.** Due to the results presented in Section 3.6, one may ask if there exist non-trivial iterative subspace trails with active S-boxes, namely P-SPN schemes for which there exist iterative subspace trails with active S-boxes but no subspace trails with inactive S-boxes or invariant subspace trails with active S-boxes. As showed in the following, such schemes exist even if they are "rare". Just to give a concrete example, consider the P-SPN over $\mathbb{F}_p^3$ (for $s = 1$ and $t = 3$) where the linear layer is defined by the matrix

$$\mathfrak{M} = \begin{pmatrix} 0 & 1 & -1 \\ 1 & -2 & 1 \\ 1 & -4 & 2 \end{pmatrix}.$$

The subspace trail

$$\left\{ \mathcal{V}_0 = \langle (1,0,0)^T \rangle, \mathcal{V}_1 = \mathfrak{M} \cdot \mathcal{V}_0 = \langle (0,1,1)^T \rangle, \mathcal{V}_2 = \mathfrak{M}^2 \cdot \mathcal{V}_0 = \langle (0,1,2)^T \rangle \right\}$$

is iterative (since $\mathcal{V}_0 = \mathfrak{M}^3 \cdot \mathcal{V}_0$) with active S-boxes. Since $\dim(\langle \mathcal{V}_0, \mathcal{V}_1, \mathcal{V}_2 \rangle) = 3$, it is not possible to set up an invariant subspace trail via the previous iterative subspace trail. Moreover, using the results and the tools presented in the paper, it is possible to show that (e.g., for the prime $p = 251$) no invariant subspace trail or trails with inactive S-boxes can cover an infinite number of rounds.

# 6 Practical Tests (Active S-Boxes)

The result given in Theorem 7 seems hard to exploit in practice: *A direct construction of the infinitely long subspace trail with active S-boxes is indeed missing.* Without such a direct construction, the computational cost of evaluating all subspaces $\mathcal{IS}$ would likely be too large, as mentioned before. Here, we solve this problem by proposing two algorithms, respectively one for the case of infinitely long invariant subspace trails and one for the case of iterated trails (both with active S-boxes). Secondly, we test several matrices over $\mathbb{F}_p$ and over $\mathbb{F}_{2^n}$ to get an idea of the percentage of "weak" matrices.

## 6.1 Algorithm for Detecting "Weak" Matrices

In the case of infinitely long subspace trails with inactive S-boxes (see Algorithm 1), the starting point is the full space $\mathbb{F}^t$: The idea is to *remove* subspaces of texts step by step until an infinitely long subspace trail with inactive S-boxes is found. Here we work in the opposite way. Exploiting the definition of $\mathcal{IS}$, the idea is to *add* subspaces until an infinitely long subspace trail with active S-boxes is found.

In the following, we assume that *no infinitely long subspace trails with inactive S-boxes exist* for the analyzed scheme. Moreover, we recall that we work under the assumption that the S-box has no linear structure. This assumption is crucial in order to have only two cases, namely the case in which the input of the S-box is constant and the case in which the input of the S-box is active (namely, the input can take any possible value). Since the S-box is a permutation, these two cases remain unchanged through the S-box. In other words, if the input is neither constant nor active, all information is lost when applying the S-box. This is not the case if the S-box has a linear structure.

### 6.1.1 Case: Infinitely Long Invariant Subspace Trails with Active S-Boxes

The algorithm we present is based on the result presented in Theorem 7 restricted to the case $l = 1$. Let $I \subseteq \{1, \ldots, s\}$ be the positions of the active S-boxes s.t. $I := \{\iota_1, \iota_2, \ldots, \iota_{|I|}\}$ (where $|I| \geq 1$). Due to the first and the second points of Theorem 7, a subspace $\mathcal{IS}$ generating an infinitely long invariant subspace trail with active S-boxes must satisfy

$$\langle e_{\iota_1}, \ldots, e_{\iota_{|I|}} \rangle \subseteq \mathcal{IS} \subseteq \langle e_{\iota_1}, \ldots, e_{\iota_{|I|}}, e_{s+1}, \ldots, e_t \rangle,$$

where remember that if $\mathcal{IS} \cap \langle e_i \rangle = \langle e_i \rangle$ and $\mathcal{IS} \cap \langle e_j \rangle = \langle e_j \rangle$, then $\langle e_i, e_j \rangle \subseteq \mathcal{IS}$ since $\mathcal{IS}$ is a subspace. Hence, if the active S-boxes are in position $I$, the subspace $\langle e_{\iota_1}, \ldots, e_{\iota_{|I|}} \rangle$ must be part of $\mathcal{IS}$. For this reason, we initialize $\mathcal{IS}$ with $\langle e_{\iota_1}, \ldots, e_{\iota_{|I|}} \rangle$.

The infinitely long subspace trail $\mathcal{IS}$ is invariant if and only if the condition $\mathcal{IS} = M \cdot \mathcal{IS}$ is satisfied. In order to achieve it, we increase the initial subspace $\mathcal{IS} \leftarrow \langle e_{\iota_1}, \ldots, e_{\iota_{|I|}} \rangle$ by adding subspaces until it stabilizes. In more detail, we compute $\langle \mathcal{IS}, M \cdot e_i, M^2 \cdot e_i, \ldots, M^j \cdot e_i, \ldots \rangle$ for each $i \in I$ and for each $j \geq 1$. Indeed, if $e_i \in \mathcal{IS}$ and if the condition $\mathcal{IS} = M \cdot \mathcal{IS}$ must be satisfied, it follows that $M \cdot e_i$ (and its entire span) is in $\mathcal{IS}$. In a similar way, if $M \cdot e_i \in \mathcal{IS}$ and if the condition $\mathcal{IS} = M \cdot \mathcal{IS}$ must be satisfied, it follows that $M \cdot (M \cdot e_i) = M^2 \cdot e_i$ (and its entire span) is in $\mathcal{IS}$, and so on. Hence, the following two scenarios are possible:

1. For each $i \in I$, there exists $j_i \geq 1$ s.t. $\langle \mathcal{IS}, M \cdot e_i, M^2 \cdot e_i, \ldots, M^{j_i} \cdot e_i \rangle = \langle \mathcal{IS}, M \cdot e_i, M^2 \cdot e_i, \ldots, M^{j_i} \cdot e_i, M^{j_i+1} \cdot e_i, M^{j_i+2} \cdot e_i, \ldots \rangle$, that is, the vectors $M^{j+h} \cdot e_i$ are already in $\mathcal{IS}$ for each $h \geq 1$. In this case, the subspace
$$\begin{aligned}\Big\langle e_{\iota_1}, M \cdot e_{\iota_1}, M^2 \cdot e_{\iota_1}, \ldots, M^j \cdot e_{\iota_1}, e_{\iota_2}, M \cdot e_{\iota_2}, M^2 \cdot e_{\iota_2}, \ldots, M^j \cdot e_{\iota_2}, \ldots, \\ e_{\iota_{|I|}}, M \cdot e_{\iota_{|I|}}, M^2 \cdot e_{\iota_{|I|}}, \ldots, M^j \cdot e_{\iota_{|I|}} \Big\rangle,\end{aligned} \tag{5}$$
where $j = \max_{i \in I}(j_i)$, is invariant under the linear layer transformation.

2. There exists no $j_i$ s.t. the previous scenario is satisfied. If this is the case, then
$$\dim(\langle \mathcal{IS}, M \cdot e_i, M^2 \cdot e_i, \ldots, M^{j_i} \cdot e_i, M^{j_i+1} \cdot e_i \rangle) = 1 + \dim(\langle \mathcal{IS}, M \cdot e_i, M^2 \cdot e_i, \ldots, M^{j_i} \cdot e_i \rangle),$$
which means that after a finite number of iterations the subspace reaches the maximum possible dimension $t$. In such a case, it follows that there exists no infinitely long invariant subspace trail with active S-boxes (apart from the trivial one) for the particular analyzed $I \subseteq \{1, \ldots, s\}$.

We emphasize the following two facts. First, it is possible that the subspace $\mathcal{IS}$ generated as in Eq. (5) does not satisfy $\mathcal{IS} \subseteq \langle e_{\iota_1}, \ldots, e_{\iota_{|I|}}, e_{s+1}, \ldots, e_t \rangle$ (that is, the active S-boxes are not only in the positions $I \subseteq \{1, \ldots, s\}$). At the same time, since Theorem 7 provides both a necessary and a sufficient condition and since we evaluate all possible subsets of $\{1, \ldots, s\}$ (corresponding to the positions of the active S-boxes), if a subspace trail is infinitely long and invariant, it is found using the previous approach.

### 6.1.2 Case: Infinitely Long Iterative Subspace Trails with Active S-Boxes

Next, we present an algorithm for the case of infinitely long iterative subspace trails with active S-boxes.

Let $\{\mathcal{V}_1, \ldots, \mathcal{V}_r\}$ be an infinitely long iterative subspace trail of period $r$. By definition, this means that it repeats itself an infinite number of times, i.e., it generates $\{\mathcal{V}_1, \mathcal{V}_2, \ldots, \mathcal{V}_r, \mathcal{V}_1, \mathcal{V}_2, \ldots, \mathcal{V}_r, \ldots, \mathcal{V}_1, \mathcal{V}_2, \ldots, \mathcal{V}_r, \ldots\}$. It is simple to observe that $\mathcal{V}_l$ is invariant every $l$ rounds for each $l \in \{1, \ldots, r\}$. This simple observation, which is satisfied by all iterative subspace trail, is the starting point for our algorithm.

Since $\mathcal{V}_l$ is invariant every $l$ rounds, the idea is to exploit a modified version of the previous algorithm (where $M$ must be replaced by $M^l$) in order to find a subspace trail with active S-boxes that is invariant every $l$ rounds (and not every single round). We call such a subspace trail an *$l$-round invariant subspace trail*. Once such an $l$-round invariant subspace trail $\mathcal{V}$ is found, the idea is simply to determine if it generates an iterative subspace trail of the form
$$\{\mathcal{V}, M \cdot \mathcal{V}, \ldots, M^{l-1} \cdot \mathcal{V}\} \tag{6}$$
with respect to Theorem 7 (namely, if there exists $\{I_0, I_1, \ldots, I_{r-1}\}$ s.t. at the $j$-th round only the $i$-th S-boxes with $i \in I_j$ are active and all the others are constant).

**Algorithm 2:** Determining the existence of (invariant) infinitely long subspace trails with *active* S-boxes based on Theorem 7.

```
Data: P-SPN scheme over $\mathbb{F}^t$ with $s$ S-boxes applied to the first $s$ words (where
      the S-box has no linear structure).
Result: 1 if (invariant) infinitely long invariant subspace trail with active S-boxes
        is found, 0 otherwise.
foreach $I_s \subseteq \{1, 2, \ldots, s\}$ s.t. $|I_s| \geq 1$ (where $I_s := \{\iota_1, \iota_2, \ldots, \iota_{|I_s|}\}$) do
    $\mathcal{IS} \leftarrow \langle e_{\iota_1}, \ldots, e_{\iota_{|I_s|}} \rangle$.
    foreach $i \in I_s$ do
        $v \leftarrow e_i$.
        do
            $\delta \leftarrow \dim(\mathcal{IS})$.
            $v \leftarrow M \cdot v$.
            $\mathcal{IS} \leftarrow \langle \mathcal{IS}, v \rangle$.
            if $\dim(\mathcal{IS}) = t$ or $\mathcal{IS} \cap \langle e_{\iota_1}, \ldots, e_{\iota_{|I_s|}}, e_{s+1}, \ldots, e_t \rangle \neq \mathcal{IS}$ then
                break (move to next $I_s$)
        while $\dim(\mathcal{IS}) > \delta$
    return 1: infinitely long invariant subspace trail with active S-boxes found: $\mathcal{IS}$
     with active S-boxes in $I_s$.
return 0: No infinitely long invariant subspace trail with active S-boxes found.
```

Note that the subspace cannot be of a generic form $\{\mathcal{V}, \mathcal{V}_2, \ldots, \mathcal{V}_l\}$ (namely, for generic subspaces $\mathcal{V}_2, \ldots, \mathcal{V}_l$), but it must be of the form just given in Eq. (6). First of all, this is due to the fact that the round function is not a random permutation, but it is defined by the linear layer $M$. Secondly, under the assumption that there is no linear structure for the S-box, it is possible to generate an infinitely long subspace trail if and only if the input of each S-box is either constant or active. In such a case, the S-box layer only changes the coset, but not the subspace itself.

### 6.1.3 Remark

Before going on, we highlight that Algorithm 2 and Algorithm 3 only look for "real" infinitely long (respectively) invariant and iterative subspace trails with active S-boxes. To be more precise, consider the case of a P-SPN scheme for which there exists both an infinitely long invariant subspace trail with active S-boxes – denoted by $\mathcal{IS}_1$ – and an infinitely long (respectively) invariant and iterative subspace trail with inactive S-boxes of period $r$ (namely, $r = 1$ for the invariant case, and $r \geq 2$ otherwise) – denoted by $\mathcal{IS}_2$ – which are independent, that is, s.t.

$$\forall j \in \{0, ..., r-1\}: \qquad \mathcal{IS}_1 \cap (M^j \cdot \mathcal{IS}_2) = \emptyset.$$

Clearly, $\langle \mathcal{IS}_1, \mathcal{IS}_2 \rangle$ is an infinitely long resp. invariant and iterative subspace trail with active S-boxes. Examples are given in Section 5.2.1, Section 5.2.1, and Appendix E.2.

At the same time, it is not hard to see that $\langle \mathcal{IS}_1, \mathcal{IS}_2 \rangle$ cannot be identified by Algorithm 3 (due to the fact that *(1)* $\mathcal{IS}_1$ is invariant, *(2)* $\mathcal{IS}_2$ is a trail with inactive S-boxes and – more importantly – *(3)* $\mathcal{IS}_1$ and $\mathcal{IS}_2$ are independent).[23] In conclusion, Algorithm 3 can only identify iterative subspace trails with active S-boxes that cannot be decomposed into *independent* subspace trails as before (e.g., an invariant one with active S-boxes and an invariant/iterative one with inactive S-boxes).

[23] In any case, note that the matrix that defines such a P-SPN scheme would be identified as weak (and so discarded) by e.g. Algorithm 2.

**Algorithm 3:** Determining the existence of (iterative) infinitely long subspace trails with *active* S-boxes of period at most $l \geq 2$ based on Theorem 7.

```
Data: P-SPN scheme over $\mathbb{F}^t$ with $s$ S-boxes applied to the first $s$ words (where
      the S-box has no linear structure).
Result: 1 if (iterative) infinitely long iterative subspace trail with active S-boxes
        (of period at most $l \geq 2$) is found, 0 otherwise.
$flag \leftarrow 0$.
$T \leftarrow \emptyset$. // $T$ stores all iterative subspace trails found
for $r \leftarrow 2$ to $l$ do
    foreach $I \subseteq \{1, 2, \ldots, s\}$ (where $I := \{\iota_1, \iota_2, \ldots, \iota_{|I|}\}$ and $I \neq \emptyset$) do
        Apply Algorithm 2 to $M^r$, and let $\mathcal{IS}$ be the resulting "invariant" subspace
          trail with active S-boxes in $I$, or let $\mathcal{IS} = \emptyset$ if such a trail does not exist.
        // Check for a meaningful iterative subspace trail
        if $\dim(\mathcal{IS}) \geq 1$ then
            if $\mathcal{IS} = M \cdot \mathcal{IS}$ then
                // The subspace trail is invariant
                break (move to next $r$)
            $I^{(1)} \leftarrow \emptyset, I^{(2)} \leftarrow \emptyset, \ldots, I^{(r-1)} \leftarrow \emptyset$.
            for $j \leftarrow 1$ to $r - 1$ do
                $\mathcal{IS} \leftarrow M \cdot \mathcal{IS}$.
                for $i \leftarrow 1$ to $s$ do
                    $\mathcal{E}^{(i)} \leftarrow \langle e_1, \ldots, e_{i-1}, e_{i+1}, \ldots, e_s, e_{s+1}, \ldots, e_t \rangle$.
                    if $\mathcal{IS} \cap \mathcal{E}^{(i)} \neq \mathcal{IS}$ (equivalently, $\mathcal{IS} \not\subseteq \mathcal{E}^{(i)}$) then
                        if $\mathcal{IS} \cap \langle e_i \rangle = \langle e_i \rangle$ then
                            $I^{(j)} \leftarrow I^{(j)} \cup \{i\}$.
                        else
                            break (move to next $r$)
            $flag \leftarrow 1$.
            $T \leftarrow T \cup \{\mathcal{IS}, r, \{I, I^{(1)}, I^{(2)}, \ldots, I^{(r-1)}\}\}$.
    // In the case $flag = 0$ (hence, $T = \emptyset$), no infinitely long
       iterative subspace trail (of period $\leq l$) was found.
    return flag: infinitely long iterative subspace trails $T$ with active S-boxes
      found.
```

### 6.1.4 Computational Costs

Here we analyze the computational cost of the two algorithms just presented.

**Cost of Algorithm 2.** We analyze the computational costs of Algorithm 2 in terms of loop iterations. First, consider the loop starting in the second line, and note that there are $2^s - 1$ non-empty subsets of $\{1, \ldots, s\}$. The second loop is iterated $|I_s|$ times for each of these subsets. For the Do-While loop, there are two possible cases. Either it finishes if the dimension of the new $\mathcal{IS}$ is equal to the dimension of the old $\mathcal{IS}$, or the dimension of $\mathcal{IS}$ increased in the last iteration. Observe that the loop ends when $\dim(\mathcal{IS}) = t$, and hence this loop is iterated at most $t - 1$ times. Consequently, the runtime of Algorithm 2 is an element in $\mathcal{O}(2^s st)$, which makes it especially efficient when using only a few S-boxes.

**Cost of Algorithm 3.** Again, we mainly focus on loop iterations for the indicator of the final cost. First, we fix the maximum period $l$ of the iterative non-invariant subspace trail. Now, Algorithm 2 is run $l - 1$ times. After that, the next loop is iterated $l' - 1$ times for

**Table 3:** Percentage of vulnerable matrices using Algorithm 1, Algorithm 2, Algorithm 3, and orders $t$, when considering prime fields $\mathrm{GF}(p)$. We denote by "S$x$" and "V$x$" the security and vulnerability w.r.t. to Algorithm $x$, respectively (e.g., S1 denotes security w.r.t. Algorithm 1, while V2 denotes vulnerability w.r.t. Algorithm 2). For Algorithm 3, we use a maximum period of $l = 2t$.

*Random Invertible*

| $\lceil \log_2(p) \rceil$ | 4 | 8 | 4 | 8 | 8 | 16 | 8 | 16 |
|---|---|---|---|---|---|---|---|---|
| $t$ | 3 | 3 | 4 | 4 | 8 | 8 | 12 | 12 |
| % (V2) | 7.70 | 0.62 | 7.80 | 0.44 | 0.34 | $< 0.01$ | 0.36 | $< 0.01$ |
| % (V2 $\wedge$ S1) | 6.50 | 0.62 | 6.96 | 0.44 | 0.32 | $< 0.01$ | 0.36 | $< 0.01$ |
| % (V2 $\vee$ V1) | 13.92 | 1.16 | 14.42 | 0.90 | 0.74 | $< 0.01$ | 0.74 | $< 0.01$ |
| % (V3) | 0.06 | $< 0.01$ | $< 0.01$ | $< 0.01$ | $< 0.01$ | $< 0.01$ | $< 0.01$ | $< 0.01$ |
| % (V3 $\wedge$ S1 $\wedge$ S2) | $< 0.01$ | $< 0.01$ | $< 0.01$ | $< 0.01$ | $< 0.01$ | $< 0.01$ | $< 0.01$ | $< 0.01$ |
| % (V3 $\vee$ V2 $\vee$ V1) | 13.92 | 1.16 | 14.42 | 0.90 | 0.74 | $< 0.01$ | 0.74 | $< 0.01$ |

*MDS (Random Cauchy)*

| $\lceil \log_2(p) \rceil$ | 4 | 8 | 4 | 8 | 8 | 16 | 8 | 16 |
|---|---|---|---|---|---|---|---|---|
| $t$ | 3 | 3 | 4 | 4 | 8 | 8 | 12 | 12 |
| % (V2) | 7.56 | 0.54 | 5.76 | 0.42 | 0.40 | $< 0.01$ | 0.76 | $< 0.01$ |
| % (V2 $\wedge$ S1) | 5.96 | 0.54 | 5.00 | 0.42 | 0.40 | $< 0.01$ | 0.76 | $< 0.01$ |
| % (V2 $\vee$ V1) | 13.68 | 1.04 | 10.76 | 1.00 | 0.90 | 0.02 | 1.14 | $< 0.01$ |
| % (V3) | $< 0.01$ | $< 0.01$ | $< 0.01$ | $< 0.01$ | $< 0.01$ | $< 0.01$ | $< 0.01$ | $< 0.01$ |
| % (V3 $\wedge$ S1 $\wedge$ S2) | $< 0.01$ | $< 0.01$ | $< 0.01$ | $< 0.01$ | $< 0.01$ | $< 0.01$ | $< 0.01$ | $< 0.01$ |
| % (V3 $\vee$ V2 $\vee$ V1) | 13.68 | 1.04 | 10.76 | 1.00 | 0.90 | 0.02 | 1.14 | $< 0.01$ |

each $l' \in \{2, \ldots, l\}$, leading to a total number of repetitions of at most $\frac{l(l+1)}{2}$. Finally, the last loop is iterated $s$ times. Operation costs inside these iterations are negligible. This leads to the total runtime being an element in $\mathcal{O}\left(l \cdot s \cdot (2^s \cdot t + l)\right)$, which again makes this algorithm especially efficient when $s$ is small.

**Computational Cost in Practice.** We used the same hardware as for the practical tests in Section 4.2, i.e., an Intel Xeon E5-2699v4 with a maximum clock frequency of 3.60 GHz. Again, we evaluate the performance of Algorithm 2 and Algorithm 3 when using matrices over prime fields and for $n = 16$, $t \in \{4, 12\}$, and $l = 2t$. For $t = 4$, Algorithm 2 takes about 3 milliseconds and Algorithm 3 takes about 40 milliseconds. For $t = 12$, Algorithm 2 takes about 16 milliseconds and Algorithm 3 takes about 1 second.

## 6.2 Percentage of "Weak" Linear Layers

Similar to the case for Algorithm 1, we estimate the percentage of "weak" linear layers with respect to Algorithm 2 and Algorithm 3. We refer to Section 4.2 for a description about the matrices we used for our tests. Again, our sample size is 50 000 and we focus on the case $s = 1$. To also get a better understanding of the differences between the results provided by our algorithms, we made the following distinctions:[24]

*(1)* matrices which are vulnerable w.r.t. Algorithm 2,

*(2)* matrices which are vulnerable w.r.t. Algorithm 2 and secure w.r.t. Algorithm 1,

[24] Naturally, the percentage of matrices fulfilling *(2)* and *(4)* has to be lower than or equal to the percentage of matrices fulfilling resp. *(1)* and *(3)* when using the same set of matrices.

**Table 4:** Percentage of vulnerable matrices using Algorithm 1, Algorithm 2, Algorithm 3, and orders $t$, when considering binary fields $GF(2^n)$. We denote by "S$x$" and "V$x$" the security and vulnerability w.r.t. to Algorithm $x$, respectively (e.g., S1 denotes security w.r.t. Algorithm 1, while V2 denotes vulnerability w.r.t. Algorithm 2). For Algorithm 3, we use a maximum period of $l = 2t$.

| *Random Invertible* | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| $n$ | 4 | 8 | 4 | 8 | 8 | 16 | 8 | 16 |
| $t$ | 3 | 3 | 4 | 4 | 8 | 8 | 12 | 12 |
| % (V2) | 6.26 | 0.40 | 6.04 | 0.36 | 0.42 | < 0.01 | 0.24 | < 0.01 |
| % (V2 $\wedge$ S1) | 5.32 | 0.40 | 5.48 | 0.34 | 0.40 | < 0.01 | 0.24 | < 0.01 |
| % (V2 $\vee$ V1) | 11.64 | 1.00 | 11.14 | 0.72 | 0.74 | < 0.01 | 0.64 | < 0.01 |
| % (V3) | 0.02 | < 0.01 | < 0.01 | < 0.01 | < 0.01 | < 0.01 | < 0.01 | < 0.01 |
| % (V3 $\wedge$ S1 $\wedge$ S2) | < 0.01 | < 0.01 | < 0.01 | < 0.01 | < 0.01 | < 0.01 | < 0.01 | < 0.01 |
| % (V3 $\vee$ V2 $\vee$ V1) | 11.64 | 1.00 | 11.14 | 0.72 | 0.74 | < 0.01 | 0.64 | < 0.01 |
| *MDS (Random Cauchy)* | | | | | | | | |
| $n$ | 4 | 8 | 4 | 8 | 8 | 16 | 8 | 16 |
| $t$ | 3 | 3 | 4 | 4 | 8 | 8 | 12 | 12 |
| % (V2) | 6.14 | 0.36 | 4.94 | 0.30 | 0.40 | < 0.01 | 0.60 | < 0.01 |
| % (V2 $\wedge$ S1) | 4.98 | 0.36 | 4.24 | 0.30 | 0.40 | < 0.01 | 0.60 | < 0.01 |
| % (V2 $\vee$ V1) | 11.58 | 0.78 | 9.62 | 0.70 | 1.06 | < 0.01 | 1.12 | < 0.01 |
| % (V3) | < 0.01 | < 0.01 | < 0.01 | < 0.01 | < 0.01 | < 0.01 | < 0.01 | < 0.01 |
| % (V3 $\wedge$ S1 $\wedge$ S2) | < 0.01 | < 0.01 | < 0.01 | < 0.01 | < 0.01 | < 0.01 | < 0.01 | < 0.01 |
| % (V3 $\vee$ V2 $\vee$ V1) | 11.58 | 0.78 | 9.62 | 0.70 | 1.06 | < 0.01 | 1.12 | < 0.01 |

*(3)* matrices which are vulnerable w.r.t. Algorithm 3, and

*(4)* matrices which are vulnerable w.r.t. Algorithm 3 and secure w.r.t. Algorithm 1 and Algorithm 2.

Table 3 and Table 4 show the results for matrices over $GF(p)$ and $GF(2^n)$ respectively. We can immediatly see that the numbers are not very different from the numbers obtained by testing Algorithm 1. Indeed, a similar amount of matrices seems to be vulnerable with respect to Algorithm 2. Interestingly, when first excluding matrices detected by Algorithm 1, the percentage is in most cases slightly lower but the difference is negligible. This hints at the fact that properties detected by Algorithm 1 and Algorithm 2 are indeed significantly different, and only one of the two algorithms is certainly not sufficient in order to find vulnerabilities.

Moreover, when looking at the numbers obtained by testing Algorithm 3, we can see how "rare" matrices are which are vulnerable w.r.t. Algorithm 3, but not vulnerable w.r.t. the other two algorithms (see also Section 5.3). Indeed, for our sample size, the percentage for such matrices was (approximately) zero.

## 6.3 Related Work

Before concluding, we mention that the approach just proposed is not completely new in the literature. In particular, a similar strategy has been proposed in [39] for the research of invariant subspace trails and in [26] for the research of weak-key subspace trails.

Let us focus on the algorithm proposed in [39]: Given an SPN-like cipher and/or SPN-like permutation, the goal is to find a subspace $\mathcal{U}$ and an offset $u$ that is invariant under the keyless round function $R(\cdot)$, namely $R(\mathcal{U} + u) = \mathcal{U} + v$ for a certain $v$. In the case of a cipher, it is sufficient to choose the round key $k \in K_{weak} = \mathcal{U} + (u - v)$ if one aims to keep the coset invariant (depending on the key schedule, such a subspace trail can cover either a finite or an infinite number of rounds).

The approach used to find such an affine subspace $\mathcal{U} + u$ is similar to the one just proposed. Starting with the smallest possible subspace (e.g., a subspace of dimension 1), the idea is to increase it until it stabilizes. In more detail, given a round function $R(\cdot)$, the algorithm guesses a starting offset for the affine subspace $\mathcal{U} + u_0$ and then maps it forwards and backwards through $R$ and $R^{-1}$, everytime computing the span of the image. If the subspace stabilizes, an invariant subspace has been found.

Both this and our approach are based on the following observation: "*Assume $u + \mathcal{A}$ is an affine subspace such that $R(u + \mathcal{A})$ is also an affine subspace $v + \mathcal{A}$. Then for any subset $X \subseteq \mathcal{A}$, the linear span of $(R(u + X) - v) \cup X$ is contained in $\mathcal{A}$*" (see [39, Lemma 1]). Hence, the algorithm presented in [39] proceeds as follows:

1. The attacker guesses one possible offset $u'$ of the affine space to be found and fixes $v' = R(u')$. Moreover, they guess a one-dimensional subspace of $\mathcal{A}$, denoted by $\mathcal{A}_0$.

2. For each $i \geq 0$, they then compute
$$\mathcal{A}_{i+1} = \langle R(\mathcal{A}_i + u') - v', \mathcal{A}_i \rangle.$$

3. If $\dim(\mathcal{A}_{i+1}) = \dim(\mathcal{A}_i)$, an invariant subspace has been found. Otherwise, if $\dim(\mathcal{A}_{i+1}) = t$ (where $R$ is defined over $\mathbb{F}^t$), the subspace is equal to the entire space. In such a case, the idea is to restart the algorithm with different offsets or a different starting subspace $\mathcal{A}_0$.

The algorithms we proposed here are similar, but some important differences can be highlighted:

- First of all, we do not work with the entire round function, but only with the matrix that defines the linear layer. This is due to the fact that we only consider partial SPN schemes for which we assume that certain S-boxes are active and others are inactive (namely, we do not consider the case of an S-box which is neither active nor constant – namely, for which the input can only take some values). Obviously, we evaluate all the $2^s$ cases of active/constant S-boxes. This is not possible in the case of an SPN scheme due to the full S-box layer. In there, one possibility is to exploit the existence of affine structures of the S-box (namely, the existence of an affine structure $\mathcal{U} + u$ s.t. S-box$(\mathcal{U} + u) = \mathcal{U} + v$).

- As a consequence, the work presented in [39] is done (in general) under the assumption of weak keys and/or weak round constants, while our work is independent of the round keys and round constants.

- The main impact of the previous facts is that the attacker must guess the initial and the final cosets in the algorithm presented in [39], while in our case the algorithm is independent of these values. Again, this is possible due to the presence of a partial S-box layer.

# 7 Conclusion and Open Problems

In this paper, we presented necessary and sufficient conditions that a (highly non-trivial) linear layer must satisfy in order to prevent the existence of infinitely long subspace trail attacks. As already mentioned, several problems are still open for future research. Besides the ones already mentioned before (e.g., the impact of considering the eigenvalues/eigenspaces of $M$ over the algebraic closure $\mathbb{F}^\star$ of $\mathbb{F}$), here we list other problems that could be interesting for future research.

- In the whole paper, we work independently of the details of the S-box, since we assume that it is not possible to set up any non-trivial subspace trail for the S-box. However, this is not always the case (e.g., consider the examples given in [40]). As a future open problem, one could extend the result given in this paper for the case of active S-boxes to this case. In particular, assume there exist non-trivial $\mathcal{U}, \mathcal{V}$ s.t. for each $u$: $S(\mathcal{U} + u) = \mathcal{V} + v$ for a certain $v$. Hence, besides the cases in which the input of the S-box is only active or constant, one should consider the case in which the input of the S-box corresponds to a coset of $\mathcal{U}$.

- It could make sense to analyze how the key schedule influences the possibility to set up a weak-key infinitely long subspace trail. What is a possible countermeasure that allows to prevent this case? Is the analysis provided in [11] valid also in the case of P-SPN schemes?

- Here, we only consider the case in which the linear layer is defined as an invertible matrix $M \in \mathbb{F}^{t\times t}$. It could be interesting to extend our results to the case in which the entries of the matrix are linearized polynomials (namely, polynomials of the form $P(x) = \bigoplus_{i=0}^{d} \rho_i \cdot x^{2^i}$ for $d \geq 1$ which can be computed efficiently over a Boolean field).

- Is it possible to extend the results presented in this paper when the scheme is defined over a ring (instead of a field)?

# References

[1] Abdelraheem, M.A., Ågren, M., Beelen, P., Leander, G.: On the Distribution of Linear Biases: Three Instructive Examples. In: Advances in Cryptology - CRYPTO 2012. LNCS, vol. 7417, pp. 50–67. Springer (2012)

[2] Albrecht, M.R., Cid, C., Grassi, L., Khovratovich, D., Lüftenegger, R., Rechberger, C., Schofnegger, M.: Algebraic Cryptanalysis of STARK-Friendly Designs: Application to MARVELlous and MiMC. In: Advances in Cryptology - ASIACRYPT 2019. LNCS, vol. 11923, pp. 371–397 (2019)

[3] Albrecht, M.R., Grassi, L., Perrin, L., Ramacher, S., Rechberger, C., Rotaru, D., Roy, A., Schofnegger, M.: Feistel structures for mpc, and more. In: Computer Security - ESORICS 2019. LNCS, vol. 11736, pp. 151–171 (2019)

[4] Albrecht, M.R., Grassi, L., Rechberger, C., Roy, A., Tiessen, T.: MiMC: Efficient Encryption and Cryptographic Hashing with Minimal Multiplicative Complexity. In: ASIACRYPT 2016. LNCS, vol. 10031, pp. 191–219 (2016)

[5] Albrecht, M.R., Rechberger, C., Schneider, T., Tiessen, T., Zohner, M.: Ciphers for MPC and FHE. In: EUROCRYPT 2015. LNCS, vol. 9056, pp. 430–454 (2015)

[6] Aly, A., Ashur, T., Ben-Sasson, E., Dhooghe, S., Szepieniec, A.: Design of Symmetric-Key Primitives for Advanced Cryptographic Protocols. Cryptology ePrint Archive, Report 2019/426 (2019), https://eprint.iacr.org/2019/426

[7] Ashur, T., Dhooghe, S.: MARVELlous: a STARK-Friendly Family of Cryptographic Primitives. Cryptology ePrint Archive, Report 2018/1098 (2018)

[8] Avanzi, R.: 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 Trans. Symmetric Cryptol. **2017**(1), 4–44 (2017)

[9] Banik, S., Bogdanov, A., Isobe, T., Shibutani, K., Hiwatari, H., Akishita, T., Regazzoni, F.: Midori: A Block Cipher for Low Energy. In: Advances in Cryptology – ASIACRYPT 2015. LNCS, vol. 9453, pp. 411–436 (2015)

[10] Bar-On, A., Dinur, I., Dunkelman, O., Lallemand, V., Keller, N., Tsaban, B.: Cryptanalysis of SP Networks with Partial Non-Linear Layers. In: EUROCRYPT 2015. LNCS, vol. 9056, pp. 315–342 (2015)

[11] Beierle, C., Canteaut, A., Leander, G., Rotella, Y.: Proving Resistance Against Invariant Attacks: How to Choose the Round Constants. In: Advances in Cryptology – CRYPTO 2017. LNCS, vol. 10402, pp. 647–678 (2017)

[12] Beyne, T.: Block Cipher Invariants as Eigenvectors of Correlation Matrices. In: Advances in Cryptology - ASIACRYPT 2018. LNCS, vol. 11272, pp. 3–31 (2018)

[13] Beyne, T., Canteaut, A., Dinur, I., Eichlseder, M., Leander, G., Leurent, G., Naya-Plasencia, M., Perrin, L., Sasaki, Y., Todo, Y., Wiemer, F.: Out of Oddity – New Cryptanalytic Techniques against Symmetric Primitives Optimized for Integrity Proof Systems. Cryptology ePrint Archive, Report 2020/188 (2020), https://eprint.iacr.org/2020/188 – accepted at Crypto 2020

[14] Biham, E., Biryukov, A., Shamir, A.: Cryptanalysis of Skipjack Reduced to 31 Rounds Using Impossible Differentials. In: EUROCRYPT 1999. LNCS, vol. 1592, pp. 12–23 (1999)

[15] Biham, E., Shamir, A.: Differential Cryptanalysis of DES-like Cryptosystems. Journal of Cryptology **4**(1), 3–72 (1991)

[16] Blondeau, C., Leander, G., Nyberg, K.: Differential-Linear Cryptanalysis Revisited. Journal of Cryptology **30**(3), 859–888 (2017)

[17] Canteaut, A., Carpov, S., Fontaine, C., Lepoint, T., Naya-Plasencia, M., Paillier, P., Sirdey, R.: Stream Ciphers: A Practical Solution for Efficient Homomorphic-Ciphertext Compression. J. Cryptology **31**(3), 885–916 (2018)

[18] Daemen, J., Govaerts, R., Vandewalle, J.: Correlation Matrices. In: Fast Software Encryption 1994 – FSE'94. LNCS, vol. 1008, pp. 275–285 (1994)

[19] Daemen, J., Rijmen, V.: The Design of Rijndael: AES - The Advanced Encryption Standard. Information Security and Cryptography, Springer (2002)

[20] Dinur, I., Kales, D., Promitzer, A., Ramacher, S., Rechberger, C.: Linear Equivalence of Block Ciphers with Partial Non-Linear Layers: Application to LowMC. In: EUROCRYPT 2019. LNCS, vol. 11476, pp. 343–372 (2019)

[21] Dinur, I., Liu, Y., Meier, W., Wang, Q.: Optimized Interpolation Attacks on LowMC. In: ASIACRYPT 2015. LNCS, vol. 9453, pp. 535–560 (2015)

[22] Dobraunig, C., Eichlseder, M., Grassi, L., Lallemand, V., Leander, G., List, E., Mendel, F., Rechberger, C.: Rasta: A Cipher with Low ANDdepth and Few ANDs per Bit. In: CRYPTO 2018. LNCS, vol. 10991, pp. 662–692 (2018)

[23] Eichlseder, M., Grassi, L., Lüftenegger, R., Øygarden, M., Rechberger, C., Schofnegger, M., Wang, Q.: An Algebraic Attack on Ciphers with Low-Degree Round Functions: Application to Full MiMC. Cryptology ePrint Archive, Report 2020/182 (2020), https://eprint.iacr.org/2020/182

[24] Gérard, B., Grosso, V., Naya-Plasencia, M., Standaert, F.X.: Block Ciphers That Are Easier to Mask: How Far Can We Go? In: CHES 2013. LNCS, vol. 8086, pp. 383–399 (2013)

[25] Grassi, L., Kales, D., Khovratovich, D., Roy, A., Rechberger, C., Schofnegger, M.: Starkad and Poseidon: New Hash Functions for Zero Knowledge Proof Systems. Cryptology ePrint Archive, Report 2019/458 (2019)

[26] Grassi, L., Leander, G., Rechberger, C., Tezcan, C., Wiemer, F.: Weak-Key Subspace Trails and Applications to AES. Cryptology ePrint Archive, Report 2019/852 (2019), https://eprint.iacr.org/2019/852

[27] Grassi, L., Lüftenegger, R., Rechberger, C., Rotaru, D., Schofnegger, M.: On a Generalization of Substitution-Permutation Networks: The HADES Design Strategy. In: Advances in Cryptology - EUROCRYPT 2020. LNCS, vol. 12106, pp. 674–704 (2020)

[28] Grassi, L., Rechberger, C., Rønjom, S.: Subspace Trail Cryptanalysis and its Applications to AES. IACR Trans. Symmetric Cryptol. **2016**(2), 192–225 (2016)

[29] Grassi, L., Rechberger, C., Rønjom, S.: A new structural-differential property of 5-round AES. In: Coron, J., Nielsen, J.B. (eds.) EUROCRYPT 2017. LNCS, vol. 10211, pp. 289–317 (2017)

[30] Grassi, L., Rechberger, C., Rotaru, D., Scholl, P., Smart, N.P.: MPC-Friendly Symmetric Key Primitives. In: ACM SIGSAC Conference on Computer and Communications Security – 2016. pp. 430–443. ACM (2016)

[31] Grosso, V., Leurent, G., Standaert, F., Varici, K.: LS-Designs: Bitslice Encryption for Efficient Masked Software Implementations. In: Fast Software Encryption - FSE 2014. LNCS, vol. 8540, pp. 18–37 (2014)

[32] Grosso, V., Standaert, F., Faust, S.: Masking vs. multiparty computation: how large is the gap for AES? J. Cryptographic Engineering **4**(1), 47–57 (2014)

[33] Huppert, B., Willems, W.: Lineare Algebra (2nd edition). Undergraduate texts in mathematics, Vieweg+Teubner, Wiesbaden (2010)

[34] Ishai, Y., Sahai, A., Wagner, D.A.: Private Circuits: Securing Hardware against Probing Attacks. In: Advances in Cryptology - CRYPTO 2003. LNCS, vol. 2729, pp. 463–481 (2003)

[35] Kales, D., Perrin, L., Promitzer, A., Ramacher, S., Rechberger, C.: Improvements to the Linear Layer of LowMC: A Faster Picnic. Cryptology ePrint Archive, Report 2017/1148 (2017)

[36] Keller, N., Rosemarin, A.: Mind the Middle Layer: The HADES Design Strategy Revisited. Cryptology ePrint Archive, Report 2020/179 (2020), https://eprint.iacr.org/2020/179

[37] Knudsen, L.R.: Truncated and Higher Order Differentials. In: FSE 1994. LNCS, vol. 1008, pp. 196–211 (1994)

[38] Leander, G., Abdelraheem, M.A., AlKhzaimi, H., Zenner, E.: A Cryptanalysis of PRINTcipher: The Invariant Subspace Attack. In: Advances in Cryptology - CRYPTO 2011. LNCS, vol. 6841, pp. 206–221 (2011)

[39] Leander, G., Minaud, B., Rønjom, S.: A Generic Approach to Invariant Subspace Attacks: Cryptanalysis of Robin, iSCREAM and Zorro. In: EUROCRYPT 2015. LNCS, vol. 9056, pp. 254–283 (2015)

[40] Leander, G., Tezcan, C., Wiemer, F.: Searching for Subspace Trails and Truncated Differentials. IACR Trans. Symmetric Cryptol. **2018**(1), 74–100 (2018)

[41] Méaux, P., Journault, A., Standaert, F., Carlet, C.: Towards Stream Ciphers for Efficient FHE with Low-Noise Ciphertexts. In: Advances in Cryptology - EUROCRYPT 2016. LNCS, vol. 9665, pp. 311–343 (2016)

[42] Piret, G., Roche, T., Carlet, C.: PICARO - A Block Cipher Allowing Efficient Higher-Order Side-Channel Resistance. In: Applied Cryptography and Network Security - ACNS 2012. LNCS, vol. 7341, pp. 311–328 (2012)

[43] Rijmen, V., Daemen, J., Preneel, B., Bosselaers, A., Win, E.D.: The Cipher SHARK. In: FSE 1996. LNCS, vol. 1039, pp. 99–111 (1996)

[44] Todo, Y., Leander, G., Sasaki, Y.: Nonlinear Invariant Attack - Practical Attack on Full SCREAM, iSCREAM, and Midori64. In: Advances in Cryptology - ASIACRYPT 2016. LNCS, vol. 10032, pp. 3–33 (2016)

[45] Wang, Y., Wu, W., Guo, Z., Yu, X.: Differential Cryptanalysis and Linear Distinguisher of Full-Round Zorro. In: ACNS 2014. LNCS, vol. 8479, pp. 308–323 (2014)

[46] Youssef, A.M., Mister, S., Tavares, S.E.: On the Design of Linear Transformations for Substitution Permutation Encryption Networks. In: Selected Areas in Cryptography - SAC 1996. pp. 40–48 (1997)

# SUPPLEMENTARY MATERIAL

## Scripts and Implementations

As supplementary material, we provide the following code files:

- `algorithms_gf2n.sage`
- `algorithms_gfp.sage`

These files contain `Sage` implementations of Algorithm 1, Algorithm 2, Algorithm 3, and various utility functions, where the first one can be used for binary fields and the second one can be used for prime fields. We also include a file named `README.txt` which contains more detailed instructions on how to use these scripts.

Further, we provide a file named `matrix_examples.txt` which contains the `Zorro` matrix (as described in Appendix D.2), and examples for the matrices described in Section 5.2.1, Section 5.3, and Appendix E.1. Note that new examples can easily be generated with our scripts, and we refer to the `README.txt` file for more details.

# A Related Work

In order to discuss the results in [1] and [12], and the relation between them and the ones presented in this paper, we first briefly recall the definition of *correlation matrices* [18].

**Definition 8.** Let $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$ be a vectorial Boolean function. The correlation matrix $C^F \in \mathbb{R}^{2^m \times 2^n}$ of $F$ is the representation of the transition matrix of $F$ with respect to the character basis of the algebra $\mathbb{C}[\mathbb{F}_2^n]$ and $\mathbb{C}[\mathbb{F}_2^m]$. The coordinates of $C^F$ are

$$C^F_{u,v} = \frac{1}{2^n} \cdot \sum_{x \in \mathbb{F}_2^n} (-1)^{u^T \cdot F(x) + v^T \cdot x}.$$

Using these notions, we recall the results presented in the literature.

**Proposition 7** (Theorem 5 of [1]). *Consider an invertible vectorial Boolean function $F$, a subspace $\mathcal{U}$, the orthogonal subspace $\mathcal{U}^\perp$, and a vector $d$. Let $C^F_{u,v}$ be the correlation matrix of $F$, and let $\omega = (\omega_u)_{u \in U^\perp}$, where $\omega_u = (-1)^{d^T \cdot u}$. Then $C^F \cdot \omega^T = \omega^T$ if and only if $F(\mathcal{U} + d) = \mathcal{U} + d$.*

This result has been generalized by Beyne in [12], who defines a "block cipher invariant" in the following way.

**Definition 9** (Definition 2 of [12]). A vector $v \in \mathbb{C}^{2^n}$ is an invariant for a block cipher $E_k : \mathbb{F}_2^n \to \mathbb{F}_2^n$ if and only if it is an eigenvector of the correlation matrix $C^{E_k}$. If $v$ is a multiple of $(1, 0, \ldots, 0)^T$, it will be called a trivial invariant.

For the case of invariant subspace trails, the same approach – *opportunely modified* – can potentially be exploited in order to find the results proposed here. In particular, using the properties of $C^F$ just presented, it follows that in the case of a round function $R_k(\cdot) = k \oplus R(\cdot) = k + \mathfrak{M} \circ \mathfrak{S}(\cdot)$, where $\mathfrak{S}(\cdot) \equiv [S(\cdot) \mid\mid \cdots \mid\mid S(\cdot) \mid\mid I(\cdot) \mid\mid \cdots \mid\mid I(\cdot)]$ and where $\mathfrak{M}(\cdot) = M \cdot (\cdot)$, it holds that

$$C^{R_k} = C^k C^R = C^k C^{\mathfrak{M}} \cdot C^{\mathfrak{S}} = C^k [C^M] \big( [C^S]^{\otimes s} \otimes [C^I]^{\otimes (t-s)} \big),$$

where $C^M_{u,v} = \delta(u + M^T \cdot v)$, $C^I_{u,v} = \delta(u + v)$, and where $C^k$ is a diagonal matrix. In the case studied here, it is not hard to see that if no S-box is active, the eigenvalues and eigenvectors of $C^M_{u,v}$ are strictly related to the eigenvalues and eigenvectors of $M$, leading to the previous result.

**Differences in Our Work.** Here we highlight the main differences in our work.

1. Both [1] and [12] focus on invariant subspaces only. As a consequence, one has to take care of the effect of the key (namely, of $C^k$) on the eigenvectors of $C^R$ (namely, of the part of the round that is independent of the key).

2. We do not require that the subspace is invariant (namely, we do not restrict ourselves to the case $R(\mathcal{U} + v) = \mathcal{U} + w$). At the same time, an $r$-round iterative subspace trail can be seen as an invariant subspace trail for $r$ rounds of the cipher. Hence, the previous result can be adapted in order to include this case.

3. In our case, we look for infinitely long iterative subspace trails in P-SPN schemes which are independent of the secret key and of the key schedule. Again, this is not possible for an SPN cipher due to the full nonlinear layer.

# B 2-Round Iterative Subspace Trail – Details

In this section, we present all the details of the concrete example of an iterative subspace trail that is not invariant given in Section 3.4.

The starting point is given by the circulant matrix $M = \text{circ}(a, b, c, d)$ with elements $a, b, c, d \in \mathbb{F}_p$, which is invertible if and only if its determinant is different from zero:

$$-a^4 + b^4 - 4ab^2c + 2a^2c^2 - c^4 + 4a^2bd + 4bc^2d - 2b^2d^2 - 4acd^2 + d^4 \neq 0 \mod p.$$

Depending on $a, b, c, d$, such a matrix can have either 2 or 4 eigenvalues and eigenvectors, while $M^2$ has always 4 eigenvalues and eigenvectors. In particular, the eigenvalues and eigenvectors of $M$ are given by

$$\begin{aligned}
\lambda_0 = a + b + c + d: &\quad (1, 1, 1, 1)^T,\\
\lambda_1 = -\sqrt{a^2 + b^2 - 2ac + c^2 - 2bd + d^2}: &\quad (b - d, -a + c + \lambda_1, d - b, a - c - \lambda_1)^T,\\
\lambda_2 = \sqrt{a^2 + b^2 - 2ac + c^2 - 2bd + d^2}: &\quad (b - d, -a + c + \lambda_2, d - b, a - c - \lambda_2)^T,\\
\lambda_3 = a - b + c - d: &\quad (1, -1, 1, -1)^T,
\end{aligned}$$

while the eigenvalues and eigenvectors of $M^2$ are given by

$$\begin{aligned}
&\Lambda_0 = (\lambda_0)^2 = a^2 + 2a(b + c + d) + b^2 + 2b(c + d) + c^2 + 2cd + d^2: \quad (1, 1, 1, 1)^T,\\
&\Lambda_1 = (\lambda_1)^2 = a^2 + b^2 - 2ac + c^2 - 2bd + d^2: \quad (1, 0, -1, 0)^T,\\
&\Lambda_2 = (\lambda_2)^2 = a^2 + b^2 - 2ac + c^2 - 2bd + d^2: \quad (0, 1, 0, -1)^T,\\
&\Lambda_3 = (\lambda_3)^2 = a^2 - 2a(b - c + d) + b^2 - 2b(c - d) + c^2 - 2cd + d^2: \quad (1, -1, 1, -1)^T.
\end{aligned}$$

Let $\mathfrak{M}_{t\times t} \in \mathbb{F}^{t\times t}$ be the matrix defined as

$$\mathfrak{M}_{5\times 5} = \begin{pmatrix} x & y_0 & y_1 & y_0 & y_1\\ z_0 & a & b & c & d\\ z_1 & b & c & d & a\\ z_2 & c & d & a & b\\ z_3 & d & a & b & c \end{pmatrix},$$

where

*(1)* the coefficients are chosen in order to provide invertibility and "full diffusion" (at word level after a finite number of rounds) for cryptographic purposes, and

*(2)* $a, b, c, d$ are chosen such that the corresponding matrix has only 2 eigenvalues, namely

$$\forall x \in \mathbb{F}_p: \qquad a^2 + b^2 - 2\cdot a\cdot c + c^2 - 2\cdot b\cdot d + d^2 \neq x^2 \mod p,$$

(remember that $x \mapsto x^2$ is not a permutation over $\mathbb{F}_p$ for a prime $p \geq 3$ – see e.g. Hermite's criterion). For example, a choice of the form $a = c$ and $b = d$ is not allowed, since the matrix would then have 4 eigenvalues.

Note that

$$\text{(1)}\qquad \underbrace{\begin{pmatrix} a & b & c & d \\ b & c & d & a \\ c & d & a & b \\ d & a & b & c \end{pmatrix}}_{\equiv \text{circ}(a,b,c,d)} \cdot \begin{pmatrix} 0 \\ 1 \\ 0 \\ -1 \end{pmatrix} = \begin{pmatrix} b-d \\ c-a \\ -(b-d) \\ -(c-a) \end{pmatrix},$$

$$\text{(2)}\qquad \begin{pmatrix} a & b & c & d \\ b & c & d & a \\ c & d & a & b \\ d & a & b & c \end{pmatrix}^2 \cdot \begin{pmatrix} 0 \\ 1 \\ 0 \\ -1 \end{pmatrix} = (a^2+b^2-2ac+c^2-2bd+d^2) \cdot \begin{pmatrix} 0 \\ 1 \\ 0 \\ -1 \end{pmatrix}, \text{ and}$$

$$\text{(3)}\qquad \begin{pmatrix} x & y & x & y \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 1 \\ 0 \\ -1 \end{pmatrix} = \begin{pmatrix} 0 \end{pmatrix}.$$

Working in $\mathbb{F}^5$, and due to these considerations, the subspace $\mathcal{S}$ defined by $\mathcal{S} = \left\langle (0,0,1,0,-1)^T \right\rangle$ is a 2-round iterative subspace trail, since

*(1)* $\mathfrak{M} \cdot \mathcal{S} = \left\langle (0, b-d, c-a, d-b, a-c)^T \right\rangle$, and

*(2)* $\mathfrak{M}^2 \cdot \mathcal{S} = \mathcal{S}$.

Finally, note that $\mathfrak{M}^2$ is not necessarily equal to a multiple of the identity. For example, note that $(\mathfrak{M}^2_{5\times 5})_{1,5} \neq 0$, where[25] $\left(\mathfrak{M}^2_{5\times 5}\right)_{1,5} = xy_0 + y_0 a + y_1 b + y_0 c + y_1 d$ is different from 0 by appropriately choosing the entries.

**Other Examples.** Note that many other examples can be constructed in a similar way. For example, the matrix $\mathfrak{M}_{8\times 8}$ defined by

$$\mathfrak{M}_{8\times 8} = \begin{pmatrix} \text{circ}(s,z,s,z) & \text{circ}(a,b,c,d) \\ \text{circ}(a,b,c,d) & \text{circ}(u,v,u,v) \end{pmatrix},$$

where $a, b, c, d$ are chosen such that the corresponding circulant matrix has only 2 eigenvalues, allows for a 2-round iterative subspace trail defined by

$$\mathcal{S} = \left\langle (0,1,0,-1,0,0,0,0)^T \right\rangle.$$

Indeed,

*(1)* $\mathfrak{M}_{8\times 8} \cdot \left\langle (0,1,0,-1,0,0,0,0)^T \right\rangle = \left\langle (0,0,0,0,b-d,c-a,d-b,a-c)^T \right\rangle$, and

*(2)* $(\mathfrak{M}_{8\times 8})^2 \cdot \left\langle (0,1,0,-1,0,0,0,0)^T \right\rangle = \left\langle (0,1,0,-1,0,0,0,0)^T \right\rangle$.

# C Truncated and Impossible Differentials

So far, we discussed the possibility to set up truncated differentials with probability 1. However, this does not guarantee security against all other generalizations, precisely truncated differentials with probability smaller than 1 and impossible differentials. Here we briefly focus on this case. However, we point out that we do not discuss the minimum number of rounds necessary to provide security against these attacks, since they strongly depend on the details of the linear layer.

[25] The entry of a matrix $M$ in the $j$-th column of the $i$-th row is denoted either by $M_{i,j}$ or by $M[i,j]$.

As we are going to show, in the case in which the details of the S-box are not taken into account, (the "basic" variants of) truncated and/or of impossible differential distinguishers which are independent of the secret key can be set up for (at most) $2R$ rounds, where $R \geq 2 \left\lfloor \frac{t-s}{s} \right\rfloor$ is the maximum number of rounds for which it is possible to set up a truncated differential with probability 1.

*Remark.* We stress that the details of the construction (e.g., the S-box, the linear layer, the key schedule) can potentially be used to improve the previous attacks. That is, $2R$ rounds refer only to the "basic" variants of such attacks, and this number must be considered only as a lower bound in order to provide security.

## C.1 Subspace Trails and Truncated Differentials

**Proposition 8.** *Given a partial SPN scheme over $\mathbb{F}^t$ with $s \leq \lceil t/2 \rceil$ S-boxes, it is always possible to set up a subspace trail with probability 1 on at least $2 \cdot \left\lfloor \frac{t-s}{s} \right\rfloor$ rounds, defined by*

$$\left\{ \underbrace{\mathcal{S}^{\left(\left\lfloor \frac{t-s}{s} \right\rfloor\right)}, M \cdot \mathcal{S}^{\left(\left\lfloor \frac{t-s}{s} \right\rfloor\right)}, \dots, M^{\left\lfloor \frac{t-s}{s} \right\rfloor - 1} \cdot \mathcal{S}^{\left(\left\lfloor \frac{t-s}{s} \right\rfloor\right)}}_{\textit{no active S-boxes}}, \mathcal{A}^{(1)}, \dots, \mathcal{A}^{\left(\left\lfloor \frac{t-s}{s} \right\rfloor\right)} \right\}, \tag{7}$$

*where $\mathcal{S}^{(\cdot)}$ is defined as in Eq. (3), where $\mathcal{A}^{(i)} := \left\langle M(e_1), \dots, M(e_s), M \cdot \mathcal{A}^{(i-1)} \right\rangle$ for each $i \geq 1$, and where $\mathcal{A}^{(0)} := M^{\left\lfloor \frac{t-s}{s} \right\rfloor - 1} \cdot \mathcal{S}^{\left(\left\lfloor \frac{t-s}{s} \right\rfloor\right)}$.*

As done before and w.l.o.g., in the following we omit the round key and constant additions (since they only change the coset and we deal with differences).

*Proof.* The subspace trail defined over the first $\left\lfloor \frac{t-s}{s} \right\rfloor$ rounds is already analyzed in Section 3.1. Such a subspace trail cannot be extended for more rounds without activating any S-box since

$$M^{\left\lfloor \frac{t-s}{s} \right\rfloor - 1} \cdot \mathcal{S}^{\left(\left\lfloor \frac{t-s}{s} \right\rfloor\right)} \not\subseteq \langle e_{s+1}, \dots, e_t \rangle .$$

Hence, at least one S-box is active after $\left\lfloor \frac{t-s}{s} \right\rfloor$ rounds. It follows that the only way to extend the trail is by increasing the dimension of such a subspace, that is,

$$R\left( M^{\left\lfloor \frac{t-s}{s} \right\rfloor} \cdot \mathcal{S}^{\left(\left\lfloor \frac{t-s}{s} \right\rfloor\right)} \right) \subseteq \mathcal{A}^{(1)} = \left\langle M^{\left\lfloor \frac{t-s}{s} \right\rfloor + 1} \cdot \mathcal{S}^{\left(\left\lfloor \frac{t-s}{s} \right\rfloor\right)}, M(e_1), \dots, M(e_s) \right\rangle .$$

Indeed, the only thing one can do is to consider the biggest subspace for which

$$\text{S-box}\left( M^{\left(\left\lfloor \frac{t-s}{s} \right\rfloor\right)} \cdot \mathcal{S}^{\left(\left\lfloor \frac{t-s}{s} \right\rfloor\right)} \right) \subseteq \left\langle \underbrace{e_1, e_2, \dots, e_s}_{\text{Due to S-boxes}}, \underbrace{M^{\left\lfloor \frac{t-s}{s} \right\rfloor} \cdot \mathcal{S}^{\left(\left\lfloor \frac{t-s}{s} \right\rfloor\right)}}_{\text{Due to identity part}} \right\rangle .$$

In this way, we lose information about the output of the S-box layer (while nothing changes for the part of the identity layer), but we can extend the subspace trail. Working in the same way, it follows that

$$R\left(\mathcal{A}^{(1)}\right) \subseteq \mathcal{A}^{(2)} = \left\langle M \cdot \mathcal{A}^{(1)}, M(e_1), \dots, M(e_s) \right\rangle ,$$

and, more generally,

$$R\left(\mathcal{A}^{(r)}\right) \subseteq \mathcal{A}^{(r+1)} = \left\langle M \cdot \mathcal{A}^{(r)}, M(e_1), \dots, M(e_s) \right\rangle .$$

This operation can be repeated until the dimension of the subspace is smaller than $t$. Since for a generic scheme the dimension of $\mathcal{S}^{\left(\left\lfloor \frac{t-s}{s} \right\rfloor\right)}$ is $s$ and the dimension increases by $s$ in each additional round, the dimension remains smaller than $t$ for up to $2 \cdot \left\lfloor \frac{t-s}{s} \right\rfloor$ rounds. □

**Truncated Differentials.** Due to the relation between subspace trails and truncated differentials [40], it is possible to set up a truncated differential distinguisher on at least $2\cdot\left\lfloor\frac{t-s}{s}\right\rfloor$ rounds with probability 1.

## C.2 Truncated Differentials with Probability Smaller than 1

Here we exploit the relation between truncated differentials and subspace trails [28, 40] and the results just given in order to analyze the minimum number of rounds to prevent these attacks. We recall following proposition from [28].

**Proposition 9.** *Let* $\left\{\mathcal{S}^{(\lfloor\frac{t-s}{s}\rfloor)}, M\cdot\mathcal{S}^{(\lfloor\frac{t-s}{s}\rfloor)},\dots, M^{\lfloor\frac{t-s}{s}\rfloor-1}\cdot\mathcal{S}^{(\lfloor\frac{t-s}{s}\rfloor)}, \mathcal{A}^{(1)},\dots,\mathcal{A}^{(\lfloor\frac{t-s}{s}\rfloor)}\right\}$ *be a subspace trail of prob. 1 defined as in Eq.* (7)*. For simplicity, let* $\mathfrak{r}=2\cdot\lfloor(t-s)/s\rfloor$ *and let*

$$\begin{aligned}\{\mathcal{V}^0,\mathcal{V}^1,\dots,\mathcal{V}^{\lfloor(t-s)/s\rfloor-1},\mathcal{V}^{\lfloor(t-s)/s\rfloor},\dots,\mathcal{V}^{2\cdot\lfloor(t-s)/s\rfloor-2}\} &:= \\ := \left\{\mathcal{S}^{(\lfloor\frac{t-s}{s}\rfloor)}, M\cdot\mathcal{S}^{(\lfloor\frac{t-s}{s}\rfloor)},\dots, M^{\lfloor\frac{t-s}{s}\rfloor-1}\cdot\mathcal{S}^{(\lfloor\frac{t-s}{s}\rfloor)}, \mathcal{A}^{(1)},\dots,\mathcal{A}^{(\lfloor\frac{t-s}{s}\rfloor)}\right\}&.\end{aligned}$$

*If there exist* $0\le v<u\le w<\mathfrak{r}$ *s.t.*

$$\frac{\dim(\mathcal{V}^v\cap\mathcal{V}^u)}{\dim(\mathcal{V}^u)}>\frac{\dim(\mathcal{V}^w)}{t}$$

*(equivalently, s.t. given a text* $x\in\mathbb{F}^t$ $P\big(x\in\mathcal{V}^v\mid x\in\mathcal{V}^u\big)>P(x\in\mathcal{V}^w\big)$*, where* $P(\cdot)$ *denotes the probability), then it is always possible to set up a truncated differential distinguisher for* $w+u-v$ *rounds with prob.* $|\mathbb{F}|^{-\dim(\mathcal{V}^u)+\dim(\mathcal{V}^v\cap\mathcal{V}^u)}$*.*

The result follows from the fact that for each pair $(x,y)$ of plaintexts, where $x\neq y$,

$$\mathrm{P}\big(E_k(x)-E_k(y)\in\mathcal{V}^w\mid x-y\in\mathcal{V}^0\big)=\mathrm{P}\big(E_k(x)-E_k(y)\in\mathcal{V}^v\mid x-y\in\mathcal{V}^u\big)=\frac{|\mathbb{F}|^{\dim(\mathcal{V}^v\cap\mathcal{V}^u)}}{|\mathbb{F}|^{\dim(\mathcal{V}^u)}}$$

independently of the secret key $k$, due to the fact that

$$\forall a,b:\quad \exists c,d \text{ s.t. } R^u(\mathcal{V}^0+a)\subseteq\mathcal{V}^u+b \text{ and } R^{w-v}(\mathcal{V}^v+b)\subseteq\mathcal{V}^w+d,$$

where $R^x(\cdot)$ denotes the $x$-round encryption function. For comparison, in the case of a random permutation $\Pi(\cdot)$,

$$\mathrm{P}\big(\Pi(x)-\Pi(y)\in\mathcal{V}^w\mid x-y\in\mathcal{V}^0\big)=\frac{|\mathbb{F}|^{\dim(\mathcal{V}^w)}}{|\mathbb{F}|^t}.$$

We finally recall that for each subspace $\mathcal{X},\mathcal{Y}$,

$$\dim(\mathcal{X}\cap\mathcal{Y})=\dim(\mathcal{X})+\dim(\mathcal{Y})-\dim(\mathcal{X}\cup\mathcal{Y}),$$

where $\dim(\mathcal{X}\cup\mathcal{Y})$ can be easily computed by using a Gram–Schmidt process on $\mathcal{X}\cup\mathcal{Y}$.

## C.3 Impossible Differentials

Impossible differential and truncated impossible differential distinguishers/attacks [14] exploit differentials that hold with probability 0.

**Proposition 10.** *Let* $\{\mathcal{V}^0,\dots,\mathcal{V}^{\mathfrak{r}-1}\}$ *be a subspace trail of prob. 1 defined as in Proposition 9. If there exist* $0\le v<u<\mathfrak{r}$ *s.t.*

$$P\big(x\in\mathcal{V}^v\mid x\in\mathcal{V}^u\big)=0$$

*(equivalently,* $\dim(\mathcal{V}^v\cap\mathcal{V}^u)=0$*), it is always possible to set up an impossible differential distinguisher for* $\mathfrak{r}+u-v$ *rounds.*

The reason of the previous result is analogous to the one given before for truncated differential distinguishers with prob. $\le 1$.

# D Results Using our Tool

## D.1 Starkad and Poseidon Matrices

In addition to the statistical tests described in Section 4, we also used our tool for the Cauchy matrices using specific starting sequences defined for Starkad and Poseidon [25]. We recall that the matrix $M'$ over $\mathbb{F}_{2^n}$ for Starkad and the matrix $M''$ over $\mathbb{F}_p$ for Poseidon are defined by

$$\mathfrak{M}'_{i,j} = \frac{1}{x_i \oplus y_j} \qquad \text{and} \qquad \mathfrak{M}''_{i,j} = \frac{1}{x_i + y_j}, \tag{8}$$

where $x_i = i$, $y_i = i + t$, and $i \in [0, t-1]$.

**Table 5:** Vulnerable matrices for Algorithm 1 and orders $t$ and field sizes $n = \lceil \log_2(p) \rceil$ when considering the Starkad and Poseidon specifications.

| Starkad Specification (over $\mathbb{F}_{2^n}$) | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| $n$ | 4 | 8 | 4 | 8 | 8 | 16 | 8 | 16 |
| $t$ | 3 | 3 | 4 | 4 | 8 | 8 | 12 | 12 |
| Vulnerable | No | No | Yes | Yes | Yes | Yes | Yes | Yes |
| **Poseidon Specification (over $\mathbb{F}_p$)** | | | | | | | | |
| $\lceil \log_2(p) \rceil$ | 4 | 8 | 4 | 8 | 8 | 16 | 8 | 16 |
| $t$ | 3 | 3 | 4 | 4 | 8 | 8 | 12 | 12 |
| Vulnerable | No | No | No | No | No | No | No | No |

**Comparison with Related Results.** When using our tool for matrices with various sizes (i.e., different values for $t$), we can observe that some matrices over $\mathbb{F}_{2^n}$ (i.e., the matrices used for Starkad) are vulnerable to the attacks described in this paper. We can also observe, however, that matrices over $\mathbb{F}_p$ using the same $t$ values are not vulnerable. The detailed results for some instances are shown in Table 5.

These results are not new in the literature, since similar conclusions have already been shown in [36, 13]. Moreover, in [36] the authors explain how to modify the choice of $x_i$ and $y_j$ in Eq. (8) in order to fix this problem. This solution consists in changing the starting sequences for the Cauchy generation method. For completeness, we also tested our algorithm for the matrices suggested in [36]. As expected, we arrive at the same conclusion, namely, that it is not possible to set up infinitely long subspace trails for the Cauchy matrices proposed in [36] (in the case of inactive S-boxes).

## D.2 Zorro Matrix

We also checked the Zorro [24] matrix with our tool. Zorro is a variant of AES where only 4 S-boxes (at the first row) are applied per round. In our setting, Zorro is a P-SPN cipher over $(\mathbb{F}_{2^8})^{16}$ with $s = 4$ where the linear layer is defined by the $16 \times 16$ matrix

$$\forall x \in (\mathbb{F}_{2^8})^{16}: \qquad M_{\texttt{Zorro}} \cdot x := MC \cdot SR \cdot x,$$

where

$$SR = \begin{pmatrix} I & 0 & 0 & 0 \\ 0 & I_2 & 0 & 0 \\ 0 & 0 & I_3 & 0 \\ 0 & 0 & 0 & I_4 \end{pmatrix},$$

where $I$ is the $4 \times 4$ identity matrix, 0 is the $4 \times 4$ null matrix and

$$I_2 = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}, \qquad I_3 = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}, \qquad I_4 = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix},$$

and where

$$MC = \begin{pmatrix} 2 \cdot I & 3 \cdot I & I & I \\ 3 \cdot I & I & I & 2 \cdot I \\ I & I & 2 \cdot I & 3 \cdot I \\ I & 2 \cdot I & 3 \cdot I & I \end{pmatrix},$$

where again $I$ is the $4 \times 4$ identity matrix, and where $2 \equiv X \in \mathbb{F}_{2^8}$ and $3 \equiv X + 1 \in \mathbb{F}_{2^8}$.

As expected, using our tool, we found that there exists no infinitely long (iterative or invariant) subspace trail for this matrix, both in the case of inactive S-boxes and in the case of active S-boxes.

# E Examples of Infinitely Long Subspace Trails with Active S-Boxes

## E.1 A Generalization of Example (4)

In Section 5.2.1, we proposed an example of a matrix for which an infinitely long invariant subspace trail with active S-boxes exists. In such an example, one entry of the matrix is fixed and equal to zero. Here we would like to show that this is not a necessary condition in order to guarantee that such subspace trails exist.

Indeed, consider again a P-SPN scheme over $\mathbb{F}^4$ with $s = 1$ (i.e., one S-box is applied in each round). Let $\mathfrak{M}$ be the matrix defined as

$$\mathfrak{M} = \begin{pmatrix} 1 & (-M_{1,3} \cdot b - M_{1,4} \cdot c)/a & M_{1,3} & M_{1,4} \\ a & (-a - M_{2,3} \cdot b - M_{2,4} \cdot c)/a & M_{2,3} & M_{2,4} \\ b & (-b - M_{3,3} \cdot b - M_{3,4} \cdot c)/a & M_{3,3} & M_{3,4} \\ c & (-c - M_{4,3} \cdot b - M_{4,4} \cdot c)/a & M_{4,3} & M_{4,4} \end{pmatrix},$$

where $a \neq 0$. A proper choice of $a, b, c$ and $\mathfrak{M}_{\cdot,\cdot}$ provides invertibility and "full diffusion" (at word level after a finite number of rounds) for cryptographic purposes.

Similar to the previous argument, it is possible to show that the subspace

$$\mathcal{S} = \left\langle e_1 = (1, 0, 0, 0)^T, v = (1, a, b, c)^T \right\rangle$$

generates an infinitely long invariant subspace trail with active S-boxes.

## E.2 Another Example of an Infinitely Long Iterative Subspace Trail with Active S-Boxes

Here, we discuss a possible construction of an iterative subspace trail with active S-boxes.

**Proposition 11.** *Given a P-SPN scheme with $s$ S-boxes defined as in Eq. (2), let $M \in \mathbb{F}^{t \times t}$ be an invertible matrix. Assume there exists an integer $l \geq 2$ s.t.*

*(1) $\lambda_1, \ldots, \lambda_\tau$ are the eigenvalues of $M$ and $\mathcal{P}_1, \ldots, \mathcal{P}_\tau$ the corresponding eigenspaces (where $\tau < t$), and*

*(2) $(\lambda_1)^l, \ldots, (\lambda_\tau)^l, \lambda_{\tau+1}, \ldots, \lambda_\psi$ where $\psi > \tau$ are the eigenvalues of $M^l$ and $\mathcal{P}_1, \ldots, \mathcal{P}_\tau$, $\mathcal{P}_{\tau+1}, \ldots, \mathcal{P}_\psi$ the corresponding eigenspaces (where $\psi \leq t$).*

*Let $I = \{i_1, \ldots, i_{|I|}\} \subseteq \{1, \ldots, s\}$ be the indices of the words with active S-boxes (where $I \neq \emptyset$). Let $\mathcal{IS} = \langle \mathcal{P}'_1, \ldots, \mathcal{P}'_\tau \rangle$ (where $\mathcal{P}'_j$ is a certain subspace of $\mathcal{P}_j$ for each $j \in \{1, \ldots, \tau\}$) be an infinitely long invariant subspace trail defined as in Theorem 5. Let $\mathcal{IS}'$ be the subspace defined as $\mathcal{IS}' = \left\langle \mathcal{S}^{(r)} \cap \mathcal{P}_{\tau+1}, \mathcal{S}^{(r)} \cap \mathcal{P}_{\tau+2}, \ldots, \mathcal{S}^{(r)} \cap \mathcal{P}_\psi \right\rangle$, where $\mathcal{S}^{(r)}$ is the subspace constructed as in Eq. (3) s.t. no S-box is active in the first $r$ rounds. If* $\dim(\mathcal{IS}'), \dim(\mathcal{IS}) \geq 1$*, the subspace (and its subspaces) defined as $\langle \mathcal{IS}, \mathcal{IS}' \rangle$ generates an infinitely long iterative (non-invariant) subspace trail with active S-boxes.*

By construction, note that the previous subspaces $\mathcal{IS}$ and $\mathcal{IS}'$ are composed of two independent parts, namely *(1)* an invariant subspace trail with active S-boxes and *(2)* an iterative subspace trails with no active S-boxes.

**Example.** In order to construct a concrete example, we combine the previous results proposed in Section 3.4 and in Section 5.2.1. Given a P-SPN scheme over $\mathbb{F}^8$ with $s = 1$, a concrete example of such a matrix is given by

$$\mathfrak{M} = \begin{pmatrix} M^{(1)} & M^{(2)} \\ M^{(3)} & M^{(4)} \end{pmatrix}$$

s.t. $\mathfrak{M}$ provides invertibility and "full diffusion" (at word level after a finite number of rounds) for cryptographic purposes, where $M^{(1)}$ is the $4 \times 4$ matrix defined in Eq. (4), $M^{(4)} = \text{circ}(a, b, c, d)$ as in Section 3.4 s.t. $\text{circ}(a, b, c, d)$ has only 2 eigenvalues, $M^{(2)}$ satisfies $M^{(2)}_{i,1} = M^{(2)}_{i,3}$ and $M^{(2)}_{i,2} = M^{(2)}_{i,4}$ for $i \in [1, 4]$, and finally $M^{(3)}$ satisfies $M^{(3)}_{i,1} = 0$ and $M^{(3)}_{i,2} + M^{(3)}_{i,3} + M^{(3)}_{i,4} = 0$ for $i \in [1, 4]$. The subspace $\mathcal{IS}$ defined as

$$\mathcal{IS} = \left\langle (1, 0, 0, 0,\ 0, 0, 0, 0)^T, \left(0, M^{(1)}_{2,0}, M^{(1)}_{3,0}, M^{(1)}_{4,0},\ 0, 0, 0, 0\right)^T, (0, 0, 0, 0,\ 0, 1, 0, -1)^T \right\rangle,$$

generates an infinitely long iterative (non-invariant) subspace trail with active S-boxes.