Hidden sums and their application on block ciphers

Carlo Brunetta, Marco Calderini, and Massimiliano Sala

Department of Mathematics, University of Trento, Italy
brunocarletta@gmail.com, marco.calderini@unitn.it, maxsalacodes@gmail.com

Abstract. We report the recent results on hidden sums obtained in the unpublished preprints by Brunetta, Calderini, and Sala. These hidden sums could be used to exploit some particular trapdoors in block ciphers. Each hidden sum is related to an elementary abelian regular subgroup. Focusing on the subgroups of the affine general linear group, we are able to characterize the maps generating these groups. From the characterization we obtain a polynomial-time algorithm to represent the elements of a binary vector space with respect to the hidden sum. Such an algorithm can be used to exploit the trapdoor in a block cipher. Then we design an efficient algorithm to perform the necessary preprocessing on the components of a cipher for the exploitation of the trapdoor.

Keywords: Cryptography, block cipher, trapdoors, hidden sums.

1 Introduction

The affine general linear group acting on a vector space is a well-understood mathematical object, for any field characteristic. Recently in an unpublished preprint [6] the authors show that some of its subgroups play an important role in cryptography. In particular, its elementary abelian regular subgroups can be exploited to insert or detect algebraic trapdoors in some block ciphers. With trapdoors we mean a hidden algebraic structure in the cipher which is known to the designer, yet unknown to anybody else, including its unfortunate legitimate users. Such a structure would allow an attacker with full knowledge to break the cipher easily, while letting the rest of the cryptographic community trust the security of the cipher.

These subgroups induce alternative operations $\circ$ on the message space V, so that $(V, \circ)$ results a vector space over $\mathbb{F}_2$. In [6], it is shown that we can use a class of these operations, called *hidden sums*, to exploit the trapdoor.

In this paper we give an overview on the results obtained in the unpublished preprints [4] and [6] regarding these hidden sums. In the first part, we report the characterization, modulo conjugation, of their elements. This characterization permits to determine a polynomial-time algorithm for representing the elements of a space $(V, \circ)$. Moreover, an attack in this context is practical. In the last part we report the study, carried out in [4], on the problem of determining the possible maps that are linear with respect to these hidden sums. More precisely, we provide an algorithm that takes as input a given linear map (with respect to the usual XOR on V) and returns the hidden sums for which this map is linear also with respect to these. Our aim is to individuate a family of hidden sums that can weaken the components of a given cipher, and to design a cipher containing the trapdoor based on hidden sums.

2 Preliminaries and notation

For any positive integer m, we let $[m] = \{1, \dots, m\}$. We write $\mathbb{F}_q$ to denote the finite field of q elements, where q is a power of prime, and $(\mathbb{F}_q)^{s\times t}$ to denote the set of all matrices with entries over $\mathbb{F}_q$ with s rows and t columns. The identity matrix of size s is denoted by I_s. We use

$$\mathbf{e}_i = (\underbrace{0,\dots,0}_{i-1}, 1, \underbrace{0,\dots,0}_{N-i}) \in (\mathbb{F}_q)^N$$

to denote the unit vector, which has a 1 in the ith position, and zeros elsewhere. Let $m \geq 1$, the vector (sub)space generated by the vectors $\mathbf{v}_1, \dots, \mathbf{v}_m$ is denoted by $\mathrm{Span}\{\mathbf{v}_1, \dots, \mathbf{v}_m\}$.

Let $V = (\mathbb{F}_q)^N$, we denote by $\mathrm{Sym}(V)$, $\mathrm{Alt}(V)$, respectively, the symmetric and the alternating group acting on V. In the following, with the symbol $+$ we refer to the usual sum over the vector space V. We denote by $T_+ = \mathrm{T}(V, +)$, $\mathrm{AGL}(V, +)$ and $\mathrm{GL}(V, +)$, respectively, the translation, affine and linear groups with respect to $+$. Moreover, the translation with respect to a vector $\mathbf{v} \in V$ will be denoted by $\sigma_{\mathbf{v}} : \mathbf{x} \mapsto \mathbf{x} + \mathbf{v}$. We write $\langle g_1, \dots, g_m\rangle$ for the group generated by $g_1, \dots, g_m$ in $\mathrm{Sym}(V)$. The map 1_V will denote the identity map on V.

Let G be a finite group acting on V. We write the action of $g \in G$ on a vector $\mathbf{v} \in V$ as $\mathbf{v}g$.

2.1 Translation based block ciphers

Most modern block ciphers are iterated ciphers, i.e. they are obtained by the composition of a finite number ℓ of rounds. Here we consider a recent definition [8] that determines a class large enough to include some common ciphers (AES [10], SERPENT [1], PRESENT [3]), but with enough algebraic structure to allow for security proofs.

Let $V = (\mathbb{F}_2)^N$ with $N = mb$, $b \geq 2$. The vector space V is a direct sum

$$V = V_1 \oplus \cdots \oplus V_b,$$

where each V_i has the same dimension m (over $\mathbb{F}_2$). For any $\mathbf{v} \in V$, we will write $\mathbf{v} = \mathbf{v}_1 \oplus \cdots \oplus \mathbf{v}_b$, where $\mathbf{v}_i \in V_i$.

Any $\gamma \in \mathrm{Sym}(V)$ that acts as $\mathbf{v}\gamma = \mathbf{v}_1\gamma_1 \oplus \cdots \oplus \mathbf{v}_b\gamma_b$, for some γ_i's in $\mathrm{Sym}(V_i)$, is a *bricklayer transformation* (a "parallel map") and any γ_i is a *brick*.

Traditionally, the maps γ_i's are called S-boxes and γ a "parallel S-box". A linear map $\lambda : V \to V$ is traditionally said a "Mixing Layer" when used in composition with parallel maps. For any $I \subset [b]$, with $I \neq \emptyset, [b]$, we say that $\bigoplus_{i\in I} V_i$ is a *wall.*

Definition 1. *A linear map $\lambda \in \mathrm{GL}(V,+)$ is a* proper mixing layer *if no wall is invariant under λ.*

We can characterize the translation-based class by the following:

Definition 2 ([8]). *A block cipher $\mathcal{C} = \{\varphi_k \mid k \in \mathcal{K}\} \subset \mathrm{Sym}(V)$, where $\mathcal{K}$ is the set containing all the session keys, over $\mathbb{F}_2$ is called* translation based (tb) *if:*

- *it is the composition of a finite number of rounds, such that any round $\rho_{k,h}$ can be written[1] as $\gamma\lambda\sigma_{\bar{k}}$, where*
  - *γ is a round-dependent bricklayer transformation (but it does not depend on k),*
  - *λ is a round-dependent linear map (but it does not depend on k),*
  - *$\bar{k}$ is in V and depends on both k and the round ($\bar{k}$ is called a "round key"),*
- *for at least one round, which we call* proper*, we have (at the same time) that λ is proper and that the map $\mathcal{K} \to V$ given by $k \mapsto \bar{k}$ is surjective.*

For a tb cipher it is possible to define the following groups. For each round h

$$\Gamma_h(\mathcal{C}) = \langle \rho_{k,h} \mid k \in \mathcal{K} \rangle \subseteq \mathrm{Sym}(V),$$

and the round function group is given by

$$\Gamma_\infty(\mathcal{C}) = \langle \Gamma_h(\mathcal{C}) \mid h = 1, \ldots, \ell \rangle.$$

An interesting problem is determining the properties of the permutation group $\Gamma_\infty(\mathcal{C}) = \Gamma_\infty$ that imply weaknesses of the cipher. A *trapdoor* is a hidden algebraic structure in the cipher which is known to the designer, yet unknown to anybody else, whose knowledge allows to obtain information on the key or to decrypt certain ciphertexts.

The first paper dealing with properties of Γ_∞ was published by Paterson [11], who showed that if this group is imprimitive, then it is possible to embed a trapdoor in the cipher. For a tb cipher in [8], the authors give sufficient conditions to guarantee that the group Γ_∞ is primitive, and the condition of proper round is crucial. However, the primitivity of Γ_∞ does not guarantee the absence of trapdoors. Indeed, if the group is contained in $\mathrm{AGL}(V,+)$ (or a conjugated of it), the encryption function is affine and once we know the image of a basis of V and the image of the zero vector, then we are able to reconstruct the matrix and the translation that compose the map.

[1] we drop the round indices

Remark 1. If T is an elementary abelian regular group, there exists a vector space structure $(V, \circ)$, where $\circ$ is the operation defined over V, such that T is the related translation group. In fact, being T regular the elements of the group can be labelled

$$T = \{\tau_{\mathbf{a}} \mid \mathbf{a} \in V\},$$

where $\tau_{\mathbf{a}}$ is the unique map in T such that $0 \mapsto \mathbf{a}$. Then, the sum between two elements is defined by $\mathbf{x} \circ \mathbf{a} := \mathbf{x}\tau_{\mathbf{a}}$. Clearly, $(V, \circ)$ is an abelian additive group and thus a vector space over $\mathbb{F}_2$.

From Remark 1, we have that there exists a copy of $\mathrm{AGL}(V, +)$ for each operation $\circ$ related to an elementary abelian regular subgroup of $\mathrm{Sym}(V)$. In the following, we use $T_\circ$, $\mathrm{AGL}(V, \circ)$ and $\mathrm{GL}(V, \circ)$ to denote, respectively, the translation, affine and linear groups corresponding to an alternative operation $\circ$.

Then, we are interesting on investigating the problem of determining whether $\Gamma_\infty(\mathcal{C})$ is contained in a group $\mathrm{AGL}(V, \circ)$ for some operation $\circ$. If the latter happens then $\circ$ is called a *hidden sum*.

Noting that the group $\Gamma_\infty(\mathcal{C})$ of a tb cipher contains the group T_+, from the principal problem above we can individuate the following problems.

1. Determine the operations $\circ$ (equivalently the translation groups $T_\circ$) such that $T_+ \subseteq \mathrm{AGL}(V, \circ)$.
2. Determine if an attack is practical whenever $\Gamma_\infty(\mathcal{C}) \subseteq \mathrm{AGL}(V, \circ)$ for a hidden sum.
3. Given a parallel S-box γ and a mixing layer λ, determine the operations $\circ$ such that γ, $\lambda \in \mathrm{AGL}(V, \circ)$ or $\gamma\lambda \in \mathrm{AGL}(V, \circ)$.

3 On hidden sum coming from subgroups of the affine linear group

In this section we investigate the elementary abelian regular subgroups of $\mathrm{AGL}(V, +)$. Using these subgroups, we present a class of hidden sums for which an attack can be practical.

The following vector space plays an important role for studying the problems above. Let T be any subgroup of the affine $\mathrm{AGL}(V, +)$ group, we can define the vector space

$$U(T) = \{\mathbf{v} \in V \mid \sigma_{\mathbf{v}} \in T\}.$$

A first results on the groups $T_\circ$ contained in $\mathrm{AGL}(V, +)$ is the following.

Proposition 1. *Let $V = (\mathbb{F}_2)^N$. Let $T_\circ \subseteq \mathrm{AGL}(V, +)$ be an elementary abelian regular subgroup. If $T_\circ \neq T_+$, then $2 - (N \mod 2) \leq \dim(U(T)) \leq N - 2$.*

Proof. See Proposition 3.6 in [6] and Proposition 3.4 in [4].

Note that for every $\mathbf{a}$, $\tau_{\mathbf{a}} \in T_\circ \subset \mathrm{AGL}(V, +)$ can be written as $\kappa_{\mathbf{a}}\sigma_{\mathbf{a}}$ for a linear map $\kappa_{\mathbf{a}} \in \mathrm{GL}(V, +)$. We will denote by $\Omega(T_\circ) = \{\kappa_{\mathbf{a}} \mid \mathbf{a} \in V\} \subset \mathrm{GL}(V, +)$. Moreover $\kappa_{\mathbf{a}} = 1_V$ if and only if $\mathbf{a} \in U(T_\circ)$. The following result characterizes the maps $\kappa_{\mathbf{a}}$ of a group $T_\circ \subset \mathrm{AGL}(V, +)$ such that $T_+ \subseteq \mathrm{AGL}(V, \circ)$, up to conjugation. These groups are of interest for Problem 1 given in Section 2.

Theorem 1. *Let $V = (\mathbb{F}_2)^{n+d}$, with $n \geq 2$, $d \geq 1$, and $T_\circ \subseteq \mathrm{AGL}(V,+)$ be such that $U(T_\circ) = \mathrm{Span}\{\mathbf{e}_{n+1}, \dots, \mathbf{e}_{n+d}\}$. Then, $T_+ \subseteq \mathrm{AGL}(V,\circ)$ if and only if for all $\kappa_\mathbf{y} \in \Omega(T_\circ)$ there exists a matrix $B_\mathbf{y} \in (\mathbb{F}_2)^{n\times d}$ such that*

$$\kappa_\mathbf{y} = \begin{bmatrix} I_n & B_\mathbf{y} \\ 0 & I_d \end{bmatrix}.$$

Proof. See Theorem 3.17 in [6].

Note that we can always suppose that $U(T_\circ)$ is generated by the last vectors of the canonical basis, as any group $T_\circ$ is conjugated, by applying a linear map, to a group $T_{\circ'}$ such that $U(T_{\circ'}) = \mathrm{Span}\{\mathbf{e}_{n+1}, \dots, \mathbf{e}_{n+d}\}$ (see [6, Theorem 3.14]).

Remark 2. When $U(T_\circ)$ is generated by the last vectors of the canonical basis, the maps $\tau_{\mathbf{e}_i}$ generate $T_\circ$, i.e. the canonical vectors form a basis also for the vector space $(V,\circ)$.

We give now a combinatorial result on the number of hidden sums contained in the affine general linear group such that $T_+ \subseteq \mathrm{AGL}(V,\circ)$ and $\dim(U(T_\circ)) = d$.

Theorem 2. *Let $N = n + d$ and*

$$\mathcal{M}_{n,d} = \{T_\circ \subseteq \mathrm{AGL}(V,+) \mid \ T_+ \subseteq \mathrm{AGL}(V,\circ) \text{ and } \dim(U(T_\circ)) = d\}.$$

Let $q = 2^d$ and define

$$\mu(n,d) = q^{\binom{n}{2}} - 1 - \sum_{r=1}^{n-2} \binom{n}{r} (q-1)^{\binom{n-r}{2}}$$

and

$$\nu(n,d) = \begin{cases} q^{\binom{n}{2}} \prod_{j=1}^{\lceil \frac{n-1}{2} \rceil} \left(1 - q^{1-2j}\right) & ,n \text{ even} \\ (q^{n-1} - 2^{n-1}) q^{\binom{n-1}{2}} \prod_{j=1}^{\lceil \frac{n-2}{2} \rceil} \left(1 - q^{1-2j}\right) & ,n \text{ odd} \end{cases}.$$

Then

$$\begin{bmatrix} N \\ d \end{bmatrix}_2 \nu(n,d) \leq |\mathcal{M}_{n,d}| \leq \begin{bmatrix} N \\ d \end{bmatrix}_2 \mu(n,d),$$

where $\left[\begin{smallmatrix} N \\ d \end{smallmatrix}\right]_q = \prod_{i=0}^{d-1} \frac{q^{N-i}-1}{q^{d-i}-1}$ is the Gaussian Binomial.

Proof. See Proposition 5.6 in [6] and Proposition 3.3 in [4].

4 Suitable hidden sums for a practical attack

In this section we want to tackle Problem 2. Thus, we want to see if given a hidden sum $\circ$ such that $\Gamma_\infty \subseteq \mathrm{AGL}(V,\circ)$, it is possible to attack the cipher. We will show that if the hidden sum is such that $T_\circ \subseteq \mathrm{AGL}(V,+)$, then a polynomial-time attack is possible.

Let $T_\circ \subseteq \mathrm{AGL}(V,+)$ be such that $T_+ \subseteq \mathrm{AGL}(V,\circ)$ (since we are supposing $T_+ \subseteq \Gamma_\infty \subseteq \mathrm{AGL}(V,\circ)$). Consider the vector space $U(T_\circ)$, which has dimension d for some $d \geq 1$. Let $g \in \mathrm{GL}(V,+)$ be such that $U(T_\circ)g = \mathrm{Span}\{\mathbf{e}_{n+1},\dots,\mathbf{e}_{n+d}\} = U(T_\diamond)$, with $T_\diamond = g^{-1}T_\circ g$. From Theorem 1 we have that the maps in $T_\diamond$ corresponding to the canonical basis are

$$\kappa_{\mathbf{e}_i}\sigma_{\mathbf{e}_i} = \begin{bmatrix} I_n & B_{\mathbf{e}_i} \\ 0 & I_d \end{bmatrix} + \mathbf{e}_i,$$

for some $B_{\mathbf{e}_i} \in (\mathbb{F}_2)^{n\times d}$. Moreover from Remark 2 we have also that $\mathbf{e}_1,\dots,\mathbf{e}_N$ is a basis of $(V,\diamond)$ and to write $\mathbf{v} \in V$ as a linear combination of these with respect to the sum $\diamond$, i.e. $\mathbf{v} = \alpha_1\mathbf{e}_1 \diamond \cdots \diamond \alpha_N\mathbf{e}_N$, we can use Algorithm 1.

```
Algorithm 1
Input: vector $\mathbf{v} = (v_1,\dots,v_N) \in V$
Output: coefficients $\alpha_1 \dots \alpha_N$.
[i] $\lambda_i \leftarrow v_i$ for $1 \leq i \leq n$;
[ii] $\mathbf{v}' \leftarrow \mathbf{v}\tau_{\mathbf{e}_1}^{\alpha_1}\cdots\tau_{\mathbf{e}_n}^{\alpha_n}$;
[iii] $\alpha_i \leftarrow v'_i$ for $n+1 \leq i \leq n+d$;
return $\alpha_1,\dots,\alpha_N$,
```

where $\tau_{\mathbf{e}_i}$ is the translation $x \mapsto x \diamond \mathbf{e}_i$ and the notation $\mathbf{x}\tau_{\mathbf{v}}^b$, with $b \in \mathbb{F}_2$, denote either $\mathbf{x}\tau_{\mathbf{v}}$ (when $b=1$) or $\mathbf{x}$ (when $b=0$). Thus, let $\mathbf{v}_i = \mathbf{e}_i g^{-1}$ for all i, applying Algorithm 1 to $\mathbf{v}g$ we can obtain the combination of $\mathbf{v}_i$'s w.r.t the sum $\circ$ of the vector $\mathbf{v}$. The complexity of this procedure is $\mathcal{O}(N^3)$. Indeed, we multiply a vector of length N for an $N \times N$ matrix (which has complexity $\mathcal{O}(N^2)$) for $n \leq N$ times.

We explain why Algorithm 1 produces the requested coefficients. Note that to find the coefficients such that $\mathbf{v} = \alpha_1\mathbf{e}_1 \diamond \cdots \diamond \alpha_N\mathbf{e}_N$ is equivalent to finding $\alpha_1,\dots,\alpha_N$ such that $\mathbf{v}\tau_{\mathbf{e}_1}^{\alpha_1}\cdots\tau_{\mathbf{e}_N}^{\alpha_N} = 0$. Now, from the form of the matrices $\kappa_{\mathbf{e}_i}$'s we can note that the first n entries of the vector $\mathbf{v}$ are left unchanged by $\kappa_{\mathbf{e}_i}$ for all i. Then, to delete a 1 in the entry $j \leq n$ of $\mathbf{v}$ we need to apply the map $\tau_{\mathbf{e}_j}$, which explains step [i]. Now, $\mathbf{v}\tau_{\mathbf{e}_1}^{\alpha_1}\cdots\tau_{\mathbf{e}_n}^{\alpha_n}$ (step [ii]) will produce a vector with the first n entries equal to 0 and, being $\tau_{\mathbf{e}_i} = \sigma_{\mathbf{e}_i}$ for $n+1 \leq i \leq n+d$, we obtain the last coefficients from step [iii].

4.1 Hidden sum attack

Let $\mathcal{C} = \{\varphi_k \mid k \in \mathcal{K}\}$ be a tb cipher such that $\Gamma_\infty \subseteq \mathrm{AGL}(V,\circ)$ for some operation $\circ$, and also $T_\circ \subseteq \mathrm{AGL}(V,+)$. Let $\dim(U(T_\circ)) = d$. Let $g \in \mathrm{GL}(V,+)$ be a linear permutation such that $U(T_\circ)g = \mathrm{Span}\{\mathbf{e}_{n+1},\dots,\mathbf{e}_{n+d}\}$. Denote by

$$[\mathbf{v}] = [\alpha_1,\dots,\alpha_N]$$

the vector with the coefficients obtained from Algorithm 1. Let $\varphi = \varphi_K$ be the encryption function, with a given unknown session key K. We are able to mount an attack, computing the matrix M and the translation vector t defining

$\varphi \in \mathrm{AGL}(V, \circ)$.
Choose the plaintext $0\varphi, \mathbf{v}_1\varphi, \ldots, \mathbf{v}_N\varphi$, where $\mathbf{v}_i = \mathbf{e}_i g^{-1}$, and compute $[0\varphi g]$, $[\mathbf{v}_1\varphi g], \ldots, [\mathbf{v}_N\varphi g]$, since the translation vector is $[t] = [0\varphi g]$ and the $[\mathbf{e}_i\varphi g]+[t]$'s are the matrix rows. In other words, we will have

$$[\mathbf{v}\varphi g] = [\mathbf{v}g] \cdot M + [t], \quad [\mathbf{v}\varphi^{-1}g] = ([\mathbf{v}g] + [t]) \cdot M^{-1},$$

for all $\mathbf{v} \in V$, where the product row by column is the standard scalar product. The knowledge of M and M^{-1} provides a global deduction (reconstruction), since it becomes trivial to encrypt and decrypt. Moreover from $[\mathbf{v}g] = [\alpha_1 \ldots, \alpha_N]$ we obtain that $\mathbf{v} = 0\tau_{\mathbf{v}_1}^{\alpha_1} \cdots \tau_{\mathbf{v}_N}^{\alpha_N}$, where $\tau_{\mathbf{v}_i} : x \mapsto x \circ \mathbf{v}_i$. So, we need only $N+1$ plaintext to reconstruct the cipher and the cost of this attack is given from the algorithm above to compute the combinations plus the cost of $N+1$ encryptions.

Our discussion has thus proved the following result.

Theorem 3. *Hidden sum trapdoors coming from translation groups such that $T_\circ \subseteq \mathrm{AGL}(V, +)$ are trapdoors, that allow for any key to perform a global deduction attack in $\mathcal{O}(N^3)$ encryptions.*

5 On hidden sums for linear maps

In this section we investigate Problem 3 given in Section 2 page 4. In particular we want to see if, for a given $\lambda \in \mathrm{GL}(V, +)$, it is possible to individuate an alternative sum $\circ$ such that $\lambda \in \mathrm{GL}(V, \circ)$.

Proposition 2. *Let $T_\circ \subseteq \mathrm{AGL}(V, +)$ and $\lambda \in \mathrm{GL}(V, +) \cap \mathrm{GL}(V, \circ)$ then $U(T_\circ)$ is invariant under the action of λ, i.e. $U(T_\circ)\lambda = U(T_\circ)$.*

Proof. See Proposition 4.1 in [4].

Proposition 3. *Let $T_\circ \subseteq \mathrm{AGL}(V, +)$ and $\lambda \in \mathrm{GL}(V, +)$. Then λ is in $\mathrm{GL}(V, +) \cap \mathrm{GL}(V, \circ)$ if and only if for all $\mathbf{x} \in V$ we have*

$$\kappa_{\mathbf{x}}\lambda = \lambda\kappa_{\mathbf{x}\lambda}, \tag{1}$$

where $\tau_{\mathbf{x}} = \kappa_{\mathbf{x}}\sigma_{\mathbf{x}}$.

Proof. See Proposition 4.2 in [4].

Now we will characterize the linear maps that are also linear for an operation $\circ$ such that $U(T_\circ)$ is generated by the last elements of the canonical basis.

Proposition 4. *Let $V = (\mathbb{F}_2)^N$, with $N = n + d$, $n \geq 2$ and $d \geq 1$. Let $T_\circ \subseteq \mathrm{AGL}(V, +)$ with $U(T_\circ) = \mathrm{Span}\{\mathbf{e}_{n+1}, ..., \mathbf{e}_{n+d}\}$. Let $\lambda \in \mathrm{GL}(V, +)$. Then $\lambda \in \mathrm{GL}(V, +) \cap \mathrm{GL}(V, \circ)$ if and only if*

$$\lambda = \begin{bmatrix} \Lambda_1 & * \\ 0 & \Lambda_2 \end{bmatrix},$$

with $\Lambda_1 \in \mathrm{GL}((\mathbb{F}_2)^n)$, $\Lambda_2 \in \mathrm{GL}((\mathbb{F}_2)^d)$, $*$ is any matrix and for all $\mathbf{x} \in V$ $B_{\mathbf{x}}\Lambda_2 = \Lambda_2 B_{\mathbf{x}\lambda}$ (see Theorem 1 for the notation of $B_{\mathbf{x}}$).

Proof. See Proposition 4.3 in [4].

Remark 3. From the propositions above, if we want to find an operation $\circ$ that linearizes a linear map $\lambda \in \mathrm{GL}(V,+)$, i.e. we want to enforce

$$\begin{bmatrix} \Lambda_1 & * \\ 0 & \Lambda_2 \end{bmatrix} \in \mathrm{GL}(V,\circ),$$

then we have to construct some matrices $B_{\mathbf{x}}$ such that $B_{\mathbf{x}}\Lambda_2 = \Lambda_1 B_{\mathbf{x}\lambda}$ for all $\mathbf{x}$. Moreover, as the standard vectors $\mathbf{e}_i$'s form a basis for the operation $\circ$, then we need to individuate only the matrices $B_{\mathbf{e}_i}$, so that $B_{\mathbf{e}_i}\Lambda_2 = \Lambda_1 B_{\mathbf{e}_i\lambda}$, and in particular that

$$B_{\mathbf{e}_i}\Lambda_2 = \Lambda_1 B_{\mathbf{e}_i\lambda} = \Lambda_1 \left(\sum_{i=1}^{n} c_i B_{\mathbf{e}_i} \right),$$

where $c_1, \ldots, c_n$ are the first components of the vector $\mathbf{e}_i\lambda$.

Algorithm 2
Input*:*

$$\lambda = \begin{bmatrix} \Lambda_1 & * \\ 0 & \Lambda_2 \end{bmatrix},$$

with $\Lambda_1 \in \mathrm{GL}((\mathbb{F}_2)^n)$*,* $\Lambda_2 \in \mathrm{GL}((\mathbb{F}_2)^d)$
Output: *all possible hidden sums such that:*

- $T_\circ \subseteq \mathrm{AGL}(V,+)$*,* $T_+ \subseteq \mathrm{AGL}(V,\circ)$*,*
- $U(T_\circ)$ *contains* $\mathbf{e}_{n+1}, ..., \mathbf{e}_{n+d}$*,*
- $\lambda \in \mathrm{GL}(V,\circ)$*.*

ALGORITHM STEPS:

I) *Consider the canonical basis* $\mathbf{e}_1, ..., \mathbf{e}_{n+d}$ *and compute* $\mathbf{e}_1\lambda, ..., \mathbf{e}_n\lambda$

II) *Solve the linear system given by the equations:*

1. *for all* $i = 1, ..., n$

$$B_{\mathbf{e}_i}\Lambda_2 = \Lambda_1 B_{\mathbf{e}_i\lambda} = \Lambda_1 \left(\sum_{i=1}^{n} c_i B_{\mathbf{e}_i} \right),$$

(where $c_1, ..., c_n$ *are the first components of the vector* $\mathbf{e}_i\lambda$*).*

2. *for all* $i = n+1, ..., n+d$

$$B_{\mathbf{e}_i} = 0,$$

3. *for all* $i = 1, ..., n$

$$\bar{\mathbf{e}}_i B_{\mathbf{e}_i} = 0,$$

(here $\bar{\mathbf{e}}_i$ *is the truncation of* $\mathbf{e}_i$ *with respect to the first* n *coordinates)*

4. *for all* $i, j = 1, ..., n$

$$\bar{\mathbf{e}}_i B_{\mathbf{e}_j} = \bar{\mathbf{e}}_j B_{\mathbf{e}_i},$$

III) return the solutions $\{B_{\mathbf{e}_i}\}_{i=1,\ldots,n+d}$.

Proposition 5. *The time complexity of Algorithm 2 is* $\mathcal{O}\left(n^6 d^3\right)$ *and the space complexity is* $\mathcal{O}\left(l \cdot 2^{d-1} n^2\right)$ *where* l *is the dimension of the solution subspace.*

In Table 1 we report some timings for different dimensions of the message space V, fixing the value of d equal to 2.

Dimension of V	n, d	Timing in seconds
64	62, 2	32.620
80	78, 2	84.380
96	94, 2	188.200
112	110, 2	338.590
128	126, 2	616.670

Table 1. Computation timing for a Mac Book Pro 15” early 2011, 4 GB Ram, Intel i7 2.00 Ghz.

In [4], we apply our algorithm to the PRESENT's mixing layer using the parameters $n = 60$ and $d = 4$. The time required to compute the operation space O is ~ 10.420 seconds and it is generated by 2360 60-tuples of 60×4 matrices. So the number of operations that linearize the mixing layer of PRESENT, is at least $|O| = 2^{2360}$.

6 Final remarks and related works

In this paper we have collected the results of two unpublished paper [4,6] about the maps generating some alternative translation groups, which may be used to embed a trapdoor in some block ciphers. We have presented a possible attack using these algebraic structures. Finally we reported an algorithm to individuate hidden sums for the mixing layer of a block cipher.

In [7] the authors give a cryptographic characteristic for the S-boxes, called Anti-crookedness (AC) in [5] and studied in [2], that permits to avoid the case $\Gamma_\infty \subseteq \mathrm{AGL}(V, \circ)$ for any operation $\circ$. On the other hand, if a permutation γ is linear with respect to some hidden sum $\circ$ then the set

$$A_\gamma = \{a \mid \mathrm{Im}(D_a\gamma) \text{ is an affine space}\} \cup \{0\}$$

contains the space $U(T_\circ)$ ($\mathrm{Im}(D_a\gamma)$ denotes the image of the derivative of γ in the direction a). In the case of the PRESENT's S-box we have that it is not AC, and for such a function the set $A_\gamma = \{(0000), (0001), (1110)\}$. Then, considering the parallel S-box γ' of PRESENT, we have $A_{\gamma'} = (A_\gamma)^{16}$. If we want to search for a possible hidden sum we need to consider all spaces candidate for $U(T_\circ)$, which means to look for all spaces that can be included in $A_{\gamma'}$. From Theorem 2, for any of these spaces we can create at least $\nu(64 - d, d)$ different hidden sums

(d is the dimension of $U(T_\circ)$). For example, taking $d = 2$, for any space $U(T_\circ)$ we have $\sim 2^{3781}$ possible hidden sums. Thus, it may be infeasible to check if a block cipher suffers of the hidden sum trapdoor, while it is easy for the designer to insert the hidden sum.

A class of operations defined here is used in [9] to weaken the nonlinearity of well-known APN S-boxes. In particular, in [9] the authors present a differential attack with respect to hidden sums.

In conclusion, we believe that the investigation of hidden sums introduced in [5] for the first time (and now well under way with many authors involved) is important both for individuating new cryptographic criteria (to design block ciphers) and for proposing new attacks on them.

References

1. R. Anderson, E. Biham, L. Knudsen, *SERPENT: A New Block Cipher Proposal*, in: Fast Software Encryption, Vol. 1372 of LNCS, Springer, 1998, pp. 222–238.
2. R. Aragona, M. Calderini, D. Maccauro, M. Sala, *On weak differential uniformity of vectorial boolean functions as a cryptographic criterion*, Applicable Algebra in Engineering, Communication and Computing 27 (5), 2016, pp. 359–372.
3. A. Bogdanov, L. R. Knudsen, G. Leander, C. Paar, A. Poschmann, M. Robshaw, Y. Seurin, C. Vikkelsoe, *PRESENT: An Ultra-Lightweight Block Cipher*, in: Proc. of CHES 2007, Vol. 7427 of LNCS, Springer, 2007, pp. 450–466.
4. C. Brunetta, M. Calderini, M. Sala, *Algorithms and bounds for hidden sums in cryptographic trapdoors*, arXiv preprint arXiv:1702.08384, 2017.
5. M. Calderini, *On Boolean functions, symmetric cryptography and algebraic coding theory*, Ph.D. thesis, University of Trento, 2015.
6. M. Calderini, M. Sala, *Elementary abelian regular subgroups as hidden sums for cryptographic trapdoors*, arXiv preprint arXiv:1702.00581, 2017.
7. A. Caranti, F. Dalla Volta, M. Sala, *An application of the O'Nan-Scott theorem to the group generated by the round functions of an AES-like cipher*, Designs, Codes and Cryptography 52 (3), 2009, pp. 293–301.
8. A. Caranti, F. Dalla Volta, M. Sala, *On some block ciphers and imprimitive groups*, AAECC 20 (5-6), 2009, pp. 229–350.
9. C. Blondeau, R. Civino and M. Sala, *Differential Attacks: Using Alternative Operations*, preprint, 2017.
10. J. Daemen, V. Rijmen, *The design of Rijndael: AES-the advanced encryption standard*, Springer Science & Business Media, 2002.
11. K. G. Paterson, *Imprimitive permutation groups and trapdoors in iterated block ciphers*, in: Fast software encryption, Vol. 1636 of LNCS, Springer, Berlin, 1999, pp. 201–214.