IACR Communications in Cryptology
ISSN 3006-5496, Vol. 2, No. 1, 23 pages.

https://doi.org/10.62056/aby7qjp10

Check for updates

Towards a Generalization of the Algebraic Attack on Stream Ciphers: A Study of the Case with Only Extremal-Degree Monomials

Pierrick Méaux[1] and Qingju Wang[2]

[1] Luxembourg University, Esch-sur-Alzette, L-4365, Luxembourg
[2] Télécom Paris, Institut Polytechnique de Paris, Palaiseau, F-91120, France

Abstract. When designing filter functions in Linear Feedback Shift Registers (LFSR) based stream ciphers, algebraic criteria of Boolean functions such as the Algebraic Immunity (AI) become key characteristics because they guarantee the security of ciphers against the powerful algebraic attacks. In this article, we abstract the algebraic attacks proposed by Courtois and Meier on filtered LFSR twenty years ago, considering how the standard algebraic attack can be generalized beyond filtered LFSR to stream ciphers that employ a Boolean filter function to an updated state. Depending on the updating process, we use different sets of annihilators than those used in the standard algebraic attack; it leads to a generalization of the concept of algebraic immunity, and in some particular cases, potentially more efficient attacks. Motivated by the filter permutator paradigm, we focus on the case where the update function is a bit-permutation, since it maintains the degree of the monomials. For example the degree of the monomials of degree up to d and from $n-d$ to n remains invariant, which leads us to consider annihilators having only monomials of these degrees. If this number of monomials is sufficiently low, linearization is feasible, allowing the linear system to be solved and revealing the key, as in the standard algebraic attack. This particular characteristic is restricted by the standard algebraic attacks and to analyze it we introduce a new notion called Extremal Algebraic Immunity (EAI).

We perform a theoretic study of the EAI criterion and explore its relation to other algebraic criteria. We prove the upper bound of the EAI of an n-variable Boolean function and further show that the EAI can be lower bounded by the AI restricted to a subset, as defined by Carlet, Méaux and Rotella at FSE 2017. We also exhibit functions with EAI guaranteed to be lower than the AI, in particular we highlight a pathological case of functions with optimal algebraic immunity and EAI only $n/4$. As applications, we determine the EAI of filter functions of some existing stream ciphers and discuss how extremal algebraic attacks using EAI could apply to variations of known ciphers.

The extremal algebraic attack does not give a better complexity than Courtois and Meier's result on the existing stream ciphers. However, we see this work as a study to avoid weaknesses in the construction of future stream ciphers.

Keywords: Algebraic immunity · Annihilators · Boolean functions · Stream ciphers

1 Introduction

The security of stream ciphers often relies on the complexity of recovering the secret key from keystream bits produced by applying a nonlinear Boolean function (called a filter) to

This is the full version of our paper published at XX
E-mail: pierrick.meaux@uni.lu (Pierrick Méaux), qingju.wang@telecom-paris.fr (Qingju Wang)

Received: 2025-01-14 Accepted: 2025-03-11

evolving internal states.

In this work, we study a specific class of algebraic attacks that exploit the structure of monomials that appear in the algebraic normal form of the annihilators of the filter's function - namely, those of extremal Hamming weight. Our motivation stems from a particular stream cipher design: the Filter Permutator (FP) paradigm, introduced with the FLIP cipher [MJSC16] for use in Hybrid Homomorphic Encryption (HHE) [NLV11]. Our proposed Extremal Algebraic Attack (EAA) is a restriction of the classical Courtois–Meier algebraic attack [CM03], tailored to settings where the updating process preserves sets of monomials with fixed Hamming weight. This condition is notably met in designs such as FLIP and (to a lesser extent) FiLIP [MCJS19b]. While the attack does not break any known cipher, it provides insight into how algebraic structure interacts with monomial stability, raising new design considerations for symmetric primitives in constrained environments like homomorphic encryption.

1.1 Filter Permutator Paradigm and Monomial Stability

The Filter Permutator paradigm, introduced in [MJSC16] (Figure 1), defines a family of stream ciphers designed to be efficiently evaluable under homomorphic encryption. Each keystream bit is obtained by applying a Boolean function f to a permuted version of the key. The permutation is publicly derived from a pseudorandom generator and varies at each round. This structure ensures that the input to the filter function always contains the same key bits, just reordered. Crucially, this means that the set of monomials with a given Hamming weight remains invariant under the updating process. A generalization of the paradigm, called Improved Filter Permutator (IFP) and associated ciphers FiLIP [MCJS19b], restricts the filter's input to a subpart of the key and adds a random whitening vector. Although this reduces the algebraic predictability of the filter input, some structural similarities remain.

These properties led us to consider a variation of the classical algebraic attack of Courtois and Meier [CM03] where, instead of targeting low-degree monomials, we focus on extremal monomials — those of very low or very high Hamming weight. This variant, which we call the Extremal Algebraic Attack (EAA), applies to FP-type designs due to their monomial stability under permutations. We consider the application of the attack on FLIP, FiLIP, variations of these schemes and of Goldreich's local pseudorandom generator [Gol00].

Nevertheless, our analysis shows that EAA does not yield more effective attacks than existing ones against published FLIP or FiLIP instances. Either the Algebraic Immunity (AI) of the filter function already leads to a better algebraic attack, or the whitening destroys the required monomial structure. However, the attack exposes a class of functions for which Extremal Algebraic Immunity (EAI) is a more meaningful criterion than classical AI.

We summarize the scope of EAA and its limitations in Section 6.3, and show that while direct attacks are limited, EAA highlights a structural vulnerability that designers should be aware of.

1.2 Courtois-Meier algebraic attack

Twenty years ago, at Eurocrypt 2003 Courtois and Meier [CM03] presented an algebraic attack on filtered Linear Feedback Shift Registers (LFSR), that broke two stream ciphers Toyocrypt and LILI-128 [SDGM00]. Throughout this paper, we call the attack (standard/classical) algebraic attack (AA). The attack impulsed a change in the design of stream ciphers, showing that using a high-degree filter function is not sufficient to prevent attacks. More precisely, the attack showed that even using a Boolean function of maximal degree, say n, as a filter, an adversary can always create an algebraic system of equations of degree at most $\lceil n/2 \rceil$ in the key variables (in a known plaintext/ciphertext attack model).

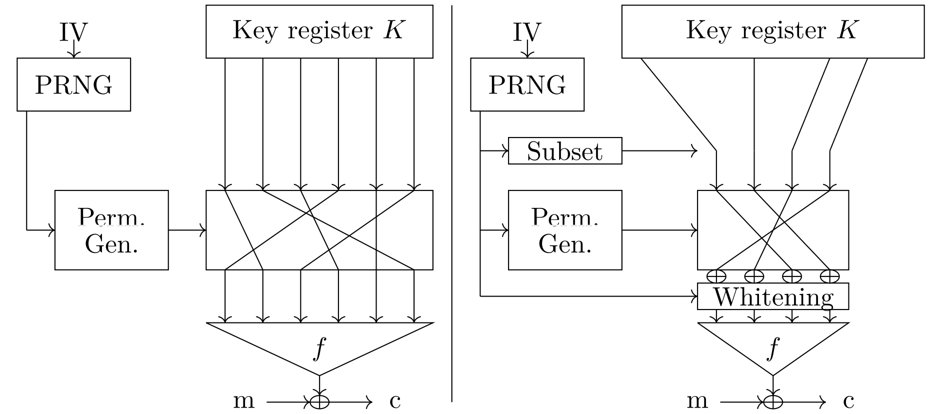


Figure 1: Filter permutator and improved filter permutator paradigms.

We recall the principle of this attack to show its generalization. First we give some necessary notations for filtered LFSRs. An LFSR is a finite state machine updated by a linear function of its previous state. It consists of a register of length N and a polynomial that defines the linear update function. The LFSR is applied to a binary key (alone or concatenated with an Initial Value) that we denote by x, hence the state of the LFSR at time i can be written as $L^{(i)}(x)$ where L is the linear transformation induced by the LFSR updating process. At time i, the filter function is applied to the LFSR state to give the i-th bit of the keystream: $s_i = f(L^{(i)}(x))$.

If we denote by d the degree of f, since L is linear, each s_i can be written as an equation of degree at most d in the key variables (composing x). The first attack considering the algebraic properties of f consists in trying to solve this algebraic system of degree d. There are many advanced approaches to solve algebraic systems over $\mathbb{F}_2$, such as Gröbner bases algorithms *e.g.* [Fau99, Fau02] or XL-algorithms [Cou02], but for simplicity of exposition we will recall the one based on *linearization*. The linearization approach treats each monomial of degree higher than one as a separate variable, and then solves the linear system newly obtained. Since there are at most $D = \sum_{j=0}^{d} \binom{n}{j} = \mathsf{D}_d^n$ monomials of degree up to d in n variables, the complexity of this attack can be estimated by $\mathcal{O}(D^\omega)$ where we denote by ω the exponent for linear algebra.

The algebraic attack proposed by Courtois and Meier [CM03] improves this complexity by not considering (the degree of) f, but the one of its products by low degree functions. This corresponds to use Boolean functions g and h of low degree such that $f \cdot g = h$. From the keystream, the adversary can derive equations of the form $s_i \cdot g(L^{(i)}(x)) = h(L^{(i)}(x))$, which are of degree at most $e = \max(\mathsf{deg}(g), \mathsf{deg}(h))$. In [CM03], the authors prove that for any function f there exist functions g and h such that $e \leq \lceil n/2 \rceil$, and $e \leq d$. This result directly leads to a linearization attack with complexity $\mathcal{O}(E^\omega)$ where $E = \sum_{j=0}^{e} \binom{n}{j} = \mathsf{D}_e^n$, giving an attack that would outperform an attack which would just consider the degree in most of the cases.

It has been shown later that finding low degree functions g and h is equivalent to finding low degree *annihilators*[1] of f or $f + 1$. The minimal value e (relatively to the function f) is in fact the minimal degree of a non null function g annihilating f or $f + 1$. Thereafter, e has been known as the notion of *algebraic immunity* [MPC04] of a Boolean function, and this parameter is the one used to bound the complexity of the algebraic attack.

[1] We say that g is an annihilator of f if $\forall x \in \mathbb{F}_2^n$, $f(x) \cdot g(x) = 0$.

1.3 Our Contributions

1.3.1 Generalizing the Courtois-Meier algebraic attack

We show how to generalize the attack in [CM03] to a larger family of stream ciphers. Instead of considering a filtered LFSR, we generalize to any binary stream cipher design defined by an updating process and a (Boolean) filter function f. We still denote x the initial state (key of the cipher), and denote by $U^{(i)}(x)$ the state at time i, obtained by applying the updating process U i times. The keystream bit s_i is obtained by applying f to $U^{(i)}(x)$. The updating process is the first part to define the attack generalization. It is a linear update L for the case of filtered LFSR, but can be quadratic for stream ciphers using Nonlinear Feedback Shift Registers (NFSR) or more complex.

The second part consists in determining subsets of monomials that appear in the Algebraic Normal Form (ANF, the representation as a multivariate polynomial over $\mathbb{F}_2$) of the annihilators of f or $f+1$. For $u \in \mathbb{F}_2^n$, we denote x^u the monomial defined by $x^u = \prod_{j\in[n]} x_j^{u_i} = \prod_{j\in\mathsf{supp}(u)} x_j$, here $[n]$ denotes the set of integers from 1 to n both included. Thereafter the sets of monomials we consider are denoted by subsets $S \subseteq \mathbb{F}_2^n$, and we focus on sets containing all the monomials appearing in the ANF of an annihilator.

Following these notations, let g be an annihilator of f, $m \in \mathbb{N}$ be the keystream size, for $i \in [m]$ when $s_i = 1$ we define S_i as the set of monomials in the ANF of $g(U^{(i)}(x))$. We define S_I as the union of the S_i for $i \in I$. When $|I| \geq |S_I|$, there are fewer monomials than equations given by the keystream, then we can apply the aforementioned linearization technique and solve the linear system to obtain the value of each monomial and then the key value. As for the algebraic attack described above, if the system is not too redundant, it gives an attack with time complexity $\mathcal{O}(|S_I|^\omega)$.

In order to further improve the efficiency of algebraic attacks, several approaches are proposed, leading to variants of algebraic attacks. One approach is to consider an annihilator h of $f+1$, to use equations when $s_i = 0$. Another approach is to take advantage of multiple linearly independent annihilators instead of one to produce more equations. For all these variants, the crucial point is the size of the union of sets where the monomials in the ANF of the updated annihilators belong to. The validity of the attack lies in the fact that the support of the ANF of the annihilators should remain in a subset of small cardinality.

The traditional attack on filtered LFSR uses that $U(x)$ is linear, therefore $g(U^{(i)}(x))$ has the same degree as $g(x)$ and therefore S_i is included in the set of monomials of degree at most $\mathsf{deg}(g)$ for all $i \in [m]$ such that $s_i = 1$. Accordingly, the subset targeted by the algebraic attack is $\{v \in \mathbb{F}_2^n \mid 0 \leq \mathsf{w_H}(v) \leq d\}$, where $\mathsf{w_H}(v)$ denotes the Hamming weight of v. Thereafter, around $\mathsf{D}^n_{\mathsf{deg}(g)}$ such keystream bits are sufficient to determine the key. Therefore, the standard algebraic attack is a particular case of the general algebraic attack we describe. In the following we introduce another particular case.

1.3.2 A study of one restriction in Courtois-Meier algebraic attack

We focus on the case where the updating process is given by a permutation of the set $[n]$, that is $U^{(i)} \in \mathcal{S}_n$ for all i. With such updating process, the sets $\mathsf{E}_{k,n} = \{v \in \mathbb{F}_2^n \mid \mathsf{w_H}(v) = k\}$ remain invariant for all $k \in [0,n]$ and we will illustrate the attack using the sets $S = \bigcup_{k\in[0,d]\cup k\in[n-d,n]} \mathsf{E}_{k,n}$, since these slices are the ones with the smallest number of elements. In this case $|S| = 2\mathsf{D}^n_d$ that enables us to compare easily with the complexity of the algebraic attack. We call this particular attack “Extremal Algebraic Attack” (EAA) and related criterion on Boolean functions the “Extremal Algebraic Immunity” (EAI) since it relies on the subsets of elements with extreme Hamming weight.

We investigate the algebraic attack given by the variant of Courtois and Meier’s attack to other sets of monomials than the one of low degree. We focus on the notion of extremal algebraic immunity, given by the sets of monomials of low (between 0 and d) and high (between $n-d$ and n) degree.

More precisely, in Section 3 we define properly the notion of extremal algebraic immunity and the set of annihilators to take in consideration for the data complexity of the attack. We describe an algorithm to compute the EAI of a Boolean function, and in the main theorem we prove an upper bound on the EAI. We also compare this upper bound to the one of the algebraic immunity (that is $\lceil n/2 \rceil$) which shows that for most Boolean functions the complexity of the EAA is lower than the one of the AA.

Then, in Section 4 and Section 5 we study cases where we can show upper bounds (respectively lower bounds) on the EAI of particular functions. In the first section we exhibit functions with EAI guaranteed to be lower than the AI. We highlight a pathological case of functions with optimal algebraic immunity and EAI only $n/4$. In Section 5 we show that the EAI can be lower bounded by the algebraic immunity restricted to a subset, as defined in [CMR17]. We generalize the result of [CMR17] on the algebraic immunity on a slice, it allows us to derive a lower bound on the EAI of functions obtained by direct sums. Additionally we exhibit a construction where the EAI and the AI of a function are the same.

Finally, in Section 6 we discuss the potential applications of the EAA. We study the value of the EAI for some functions in the literature, together with the dimension of annihilators that can be used. We also review symmetric primitives that triggered this attack generalization, and explain why it cannot apply directly. We conclude the paper in Section 7.

1.4 Related works

Other attacks relying on algebraic properties have been exhibited on filtered LFSR after Courtois and Meier's attack, such as the Fast Algebraic Attack (FAA) [Cou03] and probabilistic algebraic attack [CM03, BP05b]. The FAA considers functions g and h such that $fg = h$ but with h of higher degree than in the AA, using other techniques to cancel the high degree monomials by summing particular keystream bits. The attack relies on the relations given by the linear updating process of the LFSR. We did not find a direct relationship between the associated criterion (fast algebraic immunity) and EAI, nor works generalizing the FAA to other updating processes.

In probabilistic algebraic attacks, the attack considers a function not annihilating f on all inputs, but on most of it. In this case, there are more functions satisfying these constraints, but the algebraic system to solve is then a noisy system, where the equations are true with probability $1-\beta$ where β denotes the fraction of inputs where the product $f \cdot g$ is nonzero. The same relaxation of the annihilators is possible for EAA, directly giving probabilistic extremal algebraic attacks. We did not explore further this direction since we are not aware of concrete cryptanalyses using these approaches.

2 Preliminaries

Notations. We use $[n]$ to denote the set of integers from 1 to n both included, and $+$ instead of $\oplus$ for the addition over $\mathbb{F}_2$. For an element $v \in \mathbb{F}_2^n$ we denote by $\mathsf{w_H}(v)$ its Hamming weight $\mathsf{w_H}(v) = \#\{i \in [n] \,|\, v_i = 1\}$.

We highlight particular subsets of $\mathbb{F}_2^n$. $\mathsf{E}_{k,n}$ denotes the set $\{v \in \mathbb{F}_2^n \,|\, \mathsf{w_H}(v) = k\}$, also referred as a slice of the Boolean hypercube. We use $\mathsf{P}_{k_1,k_2,n}$ to refer to a portion of the hyper-cube, the set $\mathsf{P}_{k_1,k_2,n} = \bigcup_{k=k_1}^{k_2} \mathsf{E}_{k,n} = \{v \in \mathbb{F}_2^n \,|\, k_1 \leq \mathsf{w_H}(v) \leq k_2\}$. For these two notations we drop the n part when there is no ambiguity. We use D_d^n to denote the quantity $\sum_{i=0}^{d} \binom{n}{i}$, which is the cardinal of $\mathsf{P}_{0,d,n}$.

We use capital letters to denote matrices, such as $\mathbf{M}$. For matrices $\mathbf{A} \in \mathbb{F}_2^{\ell_1 \times c_1}$, $\mathbf{B} \in \mathbb{F}_2^{\ell_1 \times c_2}$ and $\mathbf{C} \in \mathbb{F}_2^{\ell_2 \times c_1}$, we denote the concatenation of columns of $\mathbf{A}, \mathbf{B}$ as $(\mathbf{A}|\mathbf{B}) \in \mathbb{F}_2^{\ell_1 \times (c_1+c_2)}$, and the concatenation of rows of $\mathbf{A}, \mathbf{C}$ as $(\mathbf{A}/\mathbf{C}) \in \mathbb{F}_2^{(\ell_1+\ell_2) \times c_1}$.

2.1 Boolean Functions, definitions and cryptographic criteria

In this part we provide definitions on Boolean functions and their cryptographic parameters, we refer to *e.g.* [Car21] for more details.

Definition 1 (Boolean Function)**.** A Boolean function f with n variables is a function from $\mathbb{F}_2^n$ to $\mathbb{F}_2$. The set of all Boolean functions in n variables will be denoted $\mathcal{B}_n$.

Definition 2 (Support and co-support)**.** Let f be an n-variable Boolean function, we denote by $\mathsf{supp}(f)$ its support, the set: $\{x \in \mathbb{F}_2^n \mid f(x) = 1\}$. Additionally we refer to its co-support as the set $\{x \in \mathbb{F}_2^n \mid f(x) = 0\} = \mathsf{supp}(f+1)$.

Definition 3 (Algebraic Normal Form (ANF) and degree)**.** We call Algebraic Normal Form of a Boolean function f its n-variable polynomial representation over $\mathbb{F}_2$ (*i.e.* belonging to $\mathbb{F}_2[x_1, \ldots, x_n]/\langle x_1^2 + x_1, \ldots, x_n^2 + x_n \rangle$):

$$f(x) = \sum_{I \subseteq [n]} a_I \left(\prod_{i \in I} x_i \right) = \sum_{I \subseteq [n]} a_I x^I, \qquad \text{where } a_I \in \mathbb{F}_2.$$

- The algebraic degree of f equals the global degree of its ANF: $\mathsf{deg}(f) = \max_{\{I \mid a_I = 1\}} |I|$ (with the convention that $\mathsf{deg}(0) = 0$).
- Any term $\prod_{i \in I} x_i$ in such an ANF is called a monomial and its degree equals $|I|$.

We introduce the following notations to denote sets of functions with monomials of specific degrees only.

Definition 4 (Function sets $\mathcal{F}_d$ and $\mathcal{F}_{d,n-d}$)**.** Let $n, d \in \mathbb{N}^*$ such that $d \leq n$, we denote by $\mathcal{F}_d$ and $\mathcal{F}_{d,n-d}$ the sets of Boolean functions having the following properties on their ANF coefficients $(a_I)_{I \subseteq [n]}$:

$$\mathcal{F}_d = \{f \in \mathcal{B}_n, \exists J \in \mathsf{P}_{1,d,n} \,|\, a_J = 1 \text{ and } \forall K \in \mathsf{P}_{d+1,n,n}\, a_K = 0\},$$

and for $d \leq n/2$:

$$\mathcal{F}_{d,n-d} = \{f \in \mathcal{B}_n, \exists J \in \mathsf{P}_{1,d,n} \,|\, a_J = 1 \text{ and } \forall K \in \mathsf{P}_{d+1,n-d-1,n}\, a_K = 0\}.$$

$\mathcal{F}_d$ denotes the set of functions of algebraic degree at most d (which are not constant). $\mathcal{F}_{d,n-d}$ denotes the set of non constant functions with a non empty part of degree at most d, no monomials of degree between d and $n-d-1$ and potentially monomials of degree between $n-d$ and n (which are not constant).

The following properties hold:

- for $d \leq n/2$, $\mathcal{F}_d \subsetneq \mathcal{F}_{d,n-d}$,
- $|\mathcal{F}_d| = 2^{\mathsf{D}_d^n} - 2$ and for $d < n/2$, $|\mathcal{F}_{d,n-d}| = (2^{\mathsf{D}_d^n} - 2) \cdot 2^{\mathsf{D}_d^n} = 2^{2\mathsf{D}_d^n} - 2^{\mathsf{D}_d^n + 1}$.

Definition 5 (Algebraic Immunity [MPC04])**.** The algebraic immunity of a Boolean function $f \in \mathcal{B}_n$, denoted as $\mathsf{AI}(f)$, is defined as:

$$\mathsf{AI}(f) = \min_{g \neq 0}\{\mathsf{deg}(g) \mid fg = 0 \text{ or } (f+1)g = 0\},$$

where $\mathsf{deg}(g)$ is the algebraic degree of g. The function g is called an annihilator of f (or $f+1$). Additionally we denote by $\mathsf{AN}(f) = \min_{g \neq 0}\{\mathsf{deg}(g) \mid fg = 0\}$, and by $\mathcal{D}\mathsf{AN}(f)$ the dimension of the vector space defined by the annihilators of f of degree at most $\mathsf{AI}(f)$.

We recall the generalization of algebraic immunity studied in [CMR17], named restricted algebraic immunity.

Definition 6 (Restricted Algebraic Immunity)**.** Let $n \in \mathbb{N}^*$ and $S \subseteq \mathbb{F}_2^n$, the algebraic immunity of a Boolean function $f \in \mathcal{B}_n$ restricted to the subset S, denoted as $\mathsf{AI}_S(f)$, is defined as:

$$\mathsf{AI}_S(f) = \min_{g \mid \exists y \in S,\, g(y)=1}\{\mathsf{deg}(g) \mid \forall x \in S, fg(x) = 0 \text{ or } \forall x \in S, (f+1)g = 0\}.$$

In [CMR17] the restricted AI is studied principally for the slices, *i.e.*, the subsets $\mathsf{E}_{k,n}$. In this paper we will focus on results relative to the subsets $\mathsf{P}_{0,d,n}$.

Definition 7 (Reed Muller code)**.** The Reed Muller code $\mathsf{RM}(r,n)$ is the binary code of length 2^n whose codewords are the evaluations of all Boolean functions of algebraic degree at most r in n variables on the 2^n entries.

We denote by $\mathbf{M}_{r,n}$ its generator matrix of size $\sum_{i=0}^{r}\binom{n}{i} \times 2^n$ whose term at row indexed by $u \in \mathsf{P}_{0,r,n}$ and at column indexed by $x \in \mathbb{F}_2^n$ is given by $x^u = \prod_{i=1}^{n} x_i^{u_i}$.

For a set $S \subseteq \mathbb{F}_2^n$ we denote by $\mathbf{M}_{r,n}(S)$ the matrix obtained by keeping only the columns of $\mathbf{M}_{r,n}$ whose indexes are in S.

The relations between Boolean functions and Reed Muller codes have been often used to study the properties of these objects. For example, we recall that if a vector v represents the ANF's coefficients of a degree-d function f (in the right order), then $v * \mathbf{M}_{r,n}$ is the truth table of f.

We recall a property that will be used later in the article.

Proposition 1 (Reed Muller code's property)**.** *Let $r, n \in \mathbb{N}$, such that $n > 0$ and $r \leq n$, the dimension of* $\mathsf{RM}(r,n)$ *is* D_r^n.

The algebraic immunity of a function can be determined by considering Reed Muller codes, as shown in [CM03]. The main idea consists in the following: the generator matrix of $\mathsf{RM}(r,n)$ is split in two parts, one with the columns with entries corresponding to the support of an n-variable function f, and the other corresponding to the co-support of f. Accordingly, the first matrix generates the (evaluations of the) functions fg for all g with degree at most r, and the second matrix generates the products $(f+1)g$ (recall that for an input where f takes the value 0, the product fg is 0 regardless of g, and for an input where $f = 1$, the product equals the value of g. This is why puncturing the code on the support of f yields the evaluation of fg). The rank of one of the two matrices being lower than the dimension of $\mathsf{RM}(r,n)$ is equivalent to the existence of a nonzero annihilator of degree at most r. Accordingly, the algebraic immunity of f is the smallest r such that $\mathsf{rank}(\mathbf{M}_{r,n}) \neq \mathsf{rank}(\mathbf{M}_{r,n}(\mathsf{supp}(f)))$ or $\mathsf{rank}(\mathbf{M}_{r,n}) \neq \mathsf{rank}(\mathbf{M}_{r,n}(\mathsf{supp}(f+1)))$.

We recall the secondary construction of Boolean functions called direct sum, it will be used to build examples of functions with particular parameters in the article.

Definition 8 (Direct Sum)**.** Let f be a Boolean function of n variables and g a Boolean function of m variables, f and g depending on distinct variables, the direct sum ψ of f and g is defined by:

$$\psi(x,y) = f(x) + g(y), \qquad \text{where } x \in \mathbb{F}_2^n \text{ and } y \in \mathbb{F}_2^m.$$

2.2 Symmetric Boolean functions

Symmetric Boolean functions are Boolean functions such that changing the order of the (binary) input does not change the output. Their cryptographic parameters and properties have been studied in multiple works.

Definition 9 (Symmetric Functions)**.** Let $n \in \mathbb{N}^*$, the Boolean symmetric functions are the functions which are constant on each $\mathsf{E}_{k,n}$ for $k \in [0, n]$. We focus on 2 families of symmetric functions:

- Elementary symmetric functions. Let $k \in [0, n]$, the elementary symmetric function of degree k in n variables, denoted $\sigma_{k,n}$, is the function whose ANF contains all monomials of degree k and no monomial of other degrees. When n is unambiguous from the context we denote $\sigma_{k,n}$ by σ_k.
- Threshold Functions. Let $d \in [0, n]$, the threshold function of threshold d is defined as:
$$\forall x \in \mathbb{F}_2^n, \quad \mathsf{T}_{d,n}(x) = \begin{cases} 0 & \text{if } \mathsf{w_H}(x) < d, \\ 1 & \text{otherwise.} \end{cases}$$

We will provide examples using threshold functions, we recall here some properties on elementary symmetric and threshold functions necessary for the proofs later on.

Proposition 2. *Let $n \in \mathbb{N}^*$ and $1 \leq d \leq n$ the following properties hold on symmetric functions:*

1. *Simplified representation.*

   The n-variable elementary symmetric functions form a basis of the n-variable symmetric functions, we refer to the Simplified Algebraic Normal Form (SANF) for the polynomial representation of a symmetric function as the sum of elementary symmetric functions: $f = \sum_{i=0}^{n} \lambda_i \sigma_i$, where $\lambda_i \in \mathbb{F}_2$.

2. *Product of elementary symmetric functions, e.g. [BP05a] Lemma 1.*

   Let $a, b \in \mathbb{N}$, $\sigma_a \sigma_b = \sigma_c$ where $c = bin(a) \cup bin(b)$ where $bin(\cdot)$ represents the binary decomposition ($bin(a) = (a_0, a_1, \dots, a_t)$ and $a = \sum_{i=0}^{t} a_i 2^i$).

3. *Algebraic immunity of threshold functions e.g. [CM22], Proposition 3.*

   $\mathsf{AI}(\mathsf{T}_{d,n}) = \min(d, n - d + 1)$, $\mathsf{AN}(\mathsf{T}_{d,n}) = n - d + 1$, $\mathsf{AN}(1 + \mathsf{T}_{d,n}) = d$.

4. *SANF structure of threshold functions [Méa19]:*

   - *The SANF is periodic with period $D = 2^{\lceil log(d) \rceil}$: $\forall i \in [n]$ $\lambda_i = \lambda_{i \mod D}$, where* mod D *in this context denotes the integer between* 1 *and* D *in such congruence class.*
   - *The elements in the SANF mod D belongs to an interval: $\lambda_i = 1 \Rightarrow i \mod D \in [d, D]$.*
   - *The border of the intervals are in the SANF: $\forall i \in [n]$ such that $i = d \mod D$ or $i = D \mod D$, $\lambda_i = 1$.*

3 Extremal algebraic immunity

In this section we define the extremal algebraic immunity. This criterion is designed for the case where the set defined by the union of monomials of degree from 0 to d and from $n - d$ to n is kept invariant by the updating process.

First, we define the criterion of EAI of a Boolean function , and the associated set of annihilators to take into consideration for the (data) attack complexity. Then, we exhibit the relationship between EAI and (punctured) Reed Muller codes. Finally, we prove the maximum value that the EAI can reach in the main theorem of the section, and discuss its impact on the attack compared to the standard algebraic immunity.

Definition 10 (Extremal Algebraic Immunity)**.** The extremal algebraic immunity of a Boolean function $f \in \mathcal{B}_n^*$, denoted as $\mathsf{EAI}(f)$, is defined as:

$$\mathsf{EAI}(f) = \min_{1 \le d \le n/2} \{d \mid \exists\, g \in \mathcal{F}_{d,n-d},\ fg = 0 \text{ or } (f+1)g = 0\}.$$

The EAI criterion generalizes the one of AI, instead of considering the smallest d such that f (or $f+1$) admits an annihilator in $\mathcal{F}_d$ it considers the smallest d such that $\mathcal{F}_{d,n-d}$ contains an annihilator.

Note that, by definition of $\mathcal{F}_{d,n-d}$, such annihilator have a degree at most d part which is not null. The reason to consider such annihilators, rather than the ones having only monomials of degree at least $n-d$ is to prevent to mount an attack with equations allowing to recover only the value of the high degree monomials, and not the variables. For example, the function $\prod_{i=1}^n x_i$ annihilates all functions not null in 1_n, that is half of $\mathcal{B}_n$.

For the data complexity of the extremal algebraic attack the number of annihilators of f or $f+1$ inside $\mathcal{F}_{d,n-d}$ is important since as for the algebraic attack, linearly independent annihilator can be used to produce more than one equation per keystream bit. Similarly to the $\mathcal{D}\mathsf{AN}$ for the algebraic attack (giving $2^{\mathcal{D}\mathsf{AN}(f)} - 1$ non-zero annihilators), we consider the cardinal of the set of annihilators of f that can be used for the attack.

Definition 11 (Set of usable annihilators)**.** We denote by $\mathcal{C}\mathsf{EAN}(f)$ the cardinal of the set of annihilators of f from $\mathcal{F}_{\mathsf{EAI}(f),n-\mathsf{EAI}(f)}$.

In Definition 11, the annihilators considered have at least one monomial in the part of degree up to d, since annihilators that would be null on this part lead to equations allowing to recover only the high degree monomials, as noted previously. We consider this set rather than all linear combinations obtained from the annihilators in $\mathcal{F}_{\mathsf{EAI}(f),n-\mathsf{EAI}(f)}$ since some linear combinations could have no monomials in the part of degree between 1 and $\mathsf{EAI}(f)$.

In the following proposition we exhibit the relationship between extremal algebraic immunity and (punctured) Reed-Muller codes. It generalizes the result of Courtois and Meier on the algebraic immunity. Thereafter we prove an upper bound on the EAI of any function in the main theorem of this section. First we introduce sub-matrices of the generator matrix of Reed Muller codes defined relatively to a Boolean function f.

Definition 12 (S and C matrices)**.** Let $n \in \mathbb{N}^*$ and $f \in \mathcal{B}_n$, a non constant function. We denote by $\mathbf{S} = \mathbf{M}_{n,n}(\mathsf{supp}(f))$, $\mathbf{C} = \mathbf{M}_{n,n}(\mathsf{supp}(f+1))$ and $\mathbf{S}_i^j$ (respectively $\mathbf{C}_i^j$) the sub-matrix of $\mathbf{S}$ (respectively $\mathbf{C}$) formed by the rows indexed by the monomials from degree i to j.

Proposition 3. *Let f be an n-variable Boolean function which is not constant, and the matrices $\mathbf{S}$, $\mathbf{C}$, $\mathbf{S}_i^j$ and $\mathbf{C}_i^j$ as in Definition 12. Then,* $\mathsf{EAI}(f)$ *is the smallest d such that either* $\mathsf{rank}(\mathbf{S}_0^d/\mathbf{S}_{n-d}^n) < \mathsf{D}_d^n + \mathsf{rank}(\mathbf{S}_{n-d}^n)$ *or* $\mathsf{rank}(\mathbf{C}_0^d/\mathbf{C}_{n-d}^n) < \mathsf{D}_d^n + \mathsf{rank}(\mathbf{C}_{n-d}^n)$.

Proof. We prove the statement by showing that f (respectively $(f+1)$) has an annihilator in $\mathcal{F}_{d,n-d}$ if and only if $\mathsf{rank}(\mathbf{S}_0^d/\mathbf{S}_{n-d}^n) < \mathsf{D}_d^n + \mathsf{rank}(\mathbf{S}_{n-d}^n)$ (respectively $\mathsf{rank}(\mathbf{C}_0^d/\mathbf{C}_{n-d}^n) < \mathsf{D}_d^n + \mathsf{rank}(\mathbf{C}_{n-d}^n)$). Without loss of generality we consider the case of f having such annihilator.

Assume f admits an annihilator $g \in \mathcal{F}_{d,n-d}$, then g can be written as $g_\ell + g_h$ with g_ℓ containing monomials with degree belonging to $[d]$ and g_h containing the ones of degree at least $n-d$, and g_ℓ is not null. $(g_\ell + g_h)f = 0$ therefore $g_\ell f = g_h f$, and we consider the two cases $g_\ell f = 0$ and $g_\ell f \neq 0$:

- If $g_\ell f = 0$, then a non null linear combination of the products of f by the monomials of degree at most d is giving the null function. That is, a non null linear combination of the rows of $\mathbf{S}_0^d$ gives 0_{2^n}, therefore $\mathsf{rank}(\mathbf{S}_0^d) < \sum_{i=0}^{d}\binom{n}{i} = \mathsf{D}_d^n$. Hence $\mathsf{rank}(\mathbf{S}_0^d/\mathbf{S}_{n-d}^n) < \sum_{i=0}^{d}\binom{n}{i} = \mathsf{D}_d^n + \mathsf{rank}(\mathbf{S}_{n-d}^n)$.
- If $g_\ell f \neq 0$, then a (non null) linear combination of the products of f by the monomials of degree at most d equals a (non null) linear combinations of the products of f by monomials of degree at least $n-d$. That is $\mathsf{rank}(\mathbf{S}_0^d/\mathbf{S}_{n-d}^n) < \mathsf{rank}(\mathbf{S}_0^d) + \mathsf{rank}(\mathbf{S}_{n-d}^n)$. Since $\mathsf{rank}(\mathbf{S}_0^d) \leq \mathsf{D}_d^n$, it gives the final result.

For the reverse implication, if $\mathsf{rank}(\mathbf{S}_0^d/\mathbf{S}_{n-d}^n) < \mathsf{D}_d^n + \mathsf{rank}(\mathbf{S}_{n-d}^n)$, then either $\mathsf{rank}(\mathbf{S}_0^d) < \mathsf{D}_d^n$ or $\mathsf{rank}(\mathbf{S}_0^d) = \mathsf{D}_d^n$ and there is at least a non null element belonging to the span of both matrices. In the first case it implies that f has an annihilator in $\mathcal{F}_d$ and therefore in $\mathcal{F}_{d,n-d}$. The second case implies that a linear combination of the products of f by monomials of degree at least $n-d$ give the same function as another non null combination of products of f by monomials of degree at most d. Therefore, f admits an annihilator in $\mathcal{F}_{d,n-d}$. □

Theorem 1. *Let $n \in \mathbb{N}$, $n \geq 2$ and $f \in \mathcal{B}_n$, then:*

$$\mathsf{EAI}(f) \leq \min\left(d \,\middle|\, \mathsf{D}_d^n > \frac{1}{3}\cdot 2^n\right).$$

Proof. In this proof, first using the notations from Proposition 3 we show that $\mathsf{rank}(\mathbf{S}_{n-d}^n|\mathbf{C}_{n-d}^n)$ is D_d^n and then we use it to determine a value of d such that the equalities $\mathsf{rank}(\mathbf{S}_0^d/\mathbf{S}_{n-d}^n) = \mathsf{D}_d^n + \mathsf{rank}(\mathbf{S}_{n-d}^n)$ and $\mathsf{rank}(\mathbf{C}_0^d/\mathbf{C}_{n-d}^n) = \mathsf{D}_d^n + \mathsf{rank}(\mathbf{C}_{n-d}^n)$ are not both possible.

First, we show that for $d \in [0,n]$ $\mathsf{rank}(\mathbf{S}_{n-d}^n|\mathbf{C}_{n-d}^n) = \mathsf{D}_d^n$. Since $\mathbf{S}$ and $\mathbf{C}$ are defined by the support and co-support of f, permuting the columns of $\mathbf{S}_{n-d}^n, \mathbf{C}_{n-d}^n$ we obtain $\mathbf{M}_{n-d}^n$ the sub-matrix of $\mathbf{M}_{n,n}$ restricted to the rows corresponding to degree at least $n-d$. Since $\mathsf{RM}(n,n)$ has length 2^n and dimension 2^n (Proposition 1), $\mathbf{M}_{n-d}^n$ has rank D_d^n. Thereafter we use the following fact:

$$R = \max\{\mathsf{rank}(\mathbf{S}_{n-d}^n), \mathsf{rank}(\mathbf{C}_{n-d}^n)\} = \lceil\frac{\mathsf{D}_d^n}{2}\rceil + r, \qquad (1)$$

where r is a positive integer.

Then, we derive conditions on d such that one of the two equalities cannot be satisfied anymore. Since $\mathbf{S}$ has $|\mathsf{supp}(f)|$ columns (*i.e.*, $\mathbf{S} \in \mathbb{F}_2^{2^n \times |\mathsf{supp}(f)|}$) and $\mathbf{C}$ has $|\mathsf{supp}(f+1)|$ columns (*i.e.* $\mathbf{C} \in \mathbb{F}_2^{2^n \times |\mathsf{supp}(f+1)|}$), the rank of sub-matrices obtained by these matrices is upper bounded by these quantities. Abstracting which matrix corresponds to f or $f+1$, (since $|\mathsf{supp}(f)| + |\mathsf{supp}(f+1)| = 2^n$), without loss of generality, we assume the maximum column number is the one of $\mathsf{supp}(f)$, that is $2^{n-1}+u$, and for $\mathsf{supp}(f+1)$ it is $2^{n-1}-u$ with u a positive integer no greater than 2^{n-1}. Then we consider two possibilities:

a) The biggest support is the one where the D_d^n last rows of $\mathbf{S}_{n-d}^n$ have rank R, *i.e.*, $R = \mathsf{rank}(\mathbf{S}_{n-d}^n)$, which is depicted in Figure 2a. In this case both equalities are possible only if:

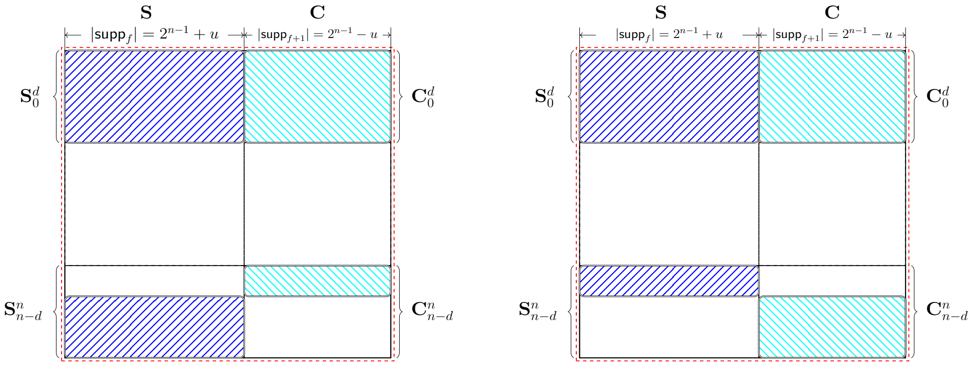
$$\mathsf{D}_d^n + R \leq 2^{n-1} + u, \text{ and } 2\mathsf{D}_d^n - R \leq 2^{n-1} - u.$$

Equation (1) implies:

$$\frac{3}{2}\mathsf{D}_d^n + r \leq 2^{n-1} + u, \text{ and } \frac{3}{2}\mathsf{D}_d^n - r \leq 2^{n-1} - u.$$

That is:

$$\frac{3}{2}\mathsf{D}_d^n \leq 2^{n-1} + (u-r), \text{ and } \frac{3}{2}\mathsf{D}_d^n \leq 2^{n-1} - (u-r).$$



(a) Case a (b) Case b

Figure 2: Two cases considered in the proof of Theorem 1

Therefore:

$$\frac{3}{2}\mathsf{D}_d^n \leq 2^{n-1} - |u-r|, \tag{2}$$

where $|\cdot|$ denotes the absolute value.

b) The smallest support is the one where the D_d^n last rows of $\mathbf{C}_{n-d}^n$ have rank R, *i.e.*, $R = \mathsf{rank}(\mathbf{C}_{n-d}^n)$, which is depicted in Figure 2b. In this case both equalities are possible only if:

$$\mathsf{D}_d^n + R \leq 2^{n-1} - u, \text{ and } 2\mathsf{D}_d^n - R \leq 2^{n-1} + u.$$

Equation (1) implies:

$$\frac{3}{2}\mathsf{D}_d^n + r \leq 2^{n-1} - u, \text{ and } \frac{3}{2}\mathsf{D}_d^n - r \leq 2^{n-1} + u.$$

That is:

$$\frac{3}{2}\mathsf{D}_d^n \leq 2^{n-1} - u - r, \text{ and } \frac{3}{2}\mathsf{D}_d^n \leq 2^{n-1} + u + r.$$

Therefore:

$$\frac{3}{2}\mathsf{D}_d^n \leq 2^{n-1} - |u+r|. \tag{3}$$

Since both u and r are positive integers, when d is such that $\mathsf{D}_d^n > \frac{1}{3}\cdot 2^n$ neither of Equations 2 and 3 holds. Thereafter, either

$$\mathsf{D}_d^n + \mathsf{rank}(\mathbf{S}_{n-d}^n) > \mathsf{rank}(\mathbf{S}) \geq \mathsf{rank}(\mathbf{S}_0^d/\mathbf{S}_{n-d}^n)$$

or

$$\mathsf{D}_d^n + \mathsf{rank}(\mathbf{C}_{n-d}^n) > \mathsf{rank}(\mathbf{C}) \geq \mathsf{rank}(\mathbf{C}_0^d/\mathbf{C}_{n-d}^n)$$

holds. Using Proposition 3, we can conclude $\mathsf{EAI}(f) \leq d$.

□

Remark 1. Note that the algebraic immunity is upper bounded by $\lceil n/2 \rceil$ as shown in [CM03]. Using the approach displayed in the proof of Theorem 1 it corresponds to the smallest d such that $\mathsf{D}_d^n > 2^{n-1} - u$. Since u is positive (null for balanced functions), the bound on the AI is the smallest d such that $\mathsf{D}_d^n > \frac{2^n}{2}$. It is to compare with $\frac{2^n}{3}$ for EAI, thereafter the upper bound on the EAI is smaller than the one on AI for all odd n greater than 1 and even n greater than 4.

The theorem shows that the EAI upper bound is smaller than the AI one, since for each function such that $2\mathsf{D}^n_{\mathsf{EAI}(f)} < \mathsf{D}^n_{\mathsf{AI}(f)}$ the EAA has a better time complexity than the standard AA we can expect the EAA to be more efficient for many functions (some examples are given in Section 4). We also remark that if other sets than $\mathsf{P}_{1,d,n} \cup \mathsf{P}_{n-d,n,n}$ are considered, for example with less slices in the the range $[n-d, n]$, Proposition 3 and Theorem 1 can also be adapted since the same arguments can be applied to other punctured Reed Muller codes.

4 Functions such that $\mathsf{EAI} \neq \mathsf{AI}$

As a preliminary remark, let us denote by e_n the bound from Theorem 1, then we obtain that for any function f such as $\mathsf{AI}(f) \geq e_n$ it holds $\mathsf{EAI}(f) < \mathsf{AI}(f)$. This is the case for all functions with optimal algebraic immunity, and in general for an overwhelming part of $\mathcal{B}_n$, since most functions have AI larger than $\frac{n}{2} - \sqrt{\frac{n}{2} \ln\left(\frac{n}{2a\ln(2)}\right)}$ for all $a < 1$ when n tends to infinity, as shown by Didier [Did06]. In the next proposition we give a different example of constructions such that $\mathsf{EAI} \neq \mathsf{AI}$.

Proposition 4. *Let $n, t \in \mathbb{N}^*$, $t \leq n/3$, and $g \in \mathcal{B}_n$ non constant such that* $\mathsf{deg}(g) < t$ *then the following holds on $f = g + \mathsf{T}_{n-t,n}$:*

$$\mathsf{EAI}(f) \leq t, \ \textit{and}\ \mathsf{AI}(f) \geq t+1.$$

Proof. First, since g has degree lower than t and is not constant, and $\mathsf{T}_{n-t,n}$ has only monomials of degree at least $n-t$ (Proposition 2 Item 4), f belongs to $\mathcal{F}_{t,n-t}$. Accordingly, $1+f$ also belongs to $\mathcal{F}_{t,n-t}$ and since $1+f$ annihilates f it guaranties $\mathsf{EAI}(f) \leq t$.

Then, we show that $\mathsf{AI}(f) \geq t+1$. We show it by contradiction. Let us assume that there exists h non null of degree at most t such that $h(\varepsilon + g + \mathsf{T}_{n-t,n}) = 0$, where $\varepsilon \in \{0,1\}$. Since g has degree lower than $t \leq n/3$ the product $h(\varepsilon + g)$ has degree lower than $2n/3$. Using Proposition 2 Item 3, $\mathsf{T}_{n-t,t}$ has no annihilator of degree lower than $t+1$, therefore the product $h(\mathsf{T}_{n-t,n})$ contains terms of degree at least $2n/3$. Therefore, $h(\varepsilon + g + \mathsf{T}_{n-t,n}) = 0$ is impossible leading to a contradiction. It allows us to conclude $\mathsf{AI}(f) \geq t+1$.

□

The gap between EAI and AI can be bigger than in the previous example, we illustrate it in the following proposition. It allows us to exhibit functions supposed to be safe against algebraic attacks, that should not be used in contexts where the extremal algebraic attack can apply.

Proposition 5. *Let $m \in \mathbb{N}^*$ and $k \in \mathbb{N}$ such that $k < 2^{m-1}$, then the threshold function $\mathsf{T}_{2^m,2^m+2k}$ is such that:*

$$\mathsf{EAI}(\mathsf{T}_{2^m,2^m+2k}) = k, \ \textit{and}\ \mathsf{AI}(\mathsf{T}_{2^m,2^m+2k}) = 2k+1.$$

Furthermore,

$$\mathsf{EAI}(\mathsf{T}_{2^m,2^m+2k+1}) = k+1, \ \textit{and}\ \mathsf{AI}(\mathsf{T}_{2^m,2^m+2k+1}) = 2k+2.$$

Proof. First, we obtain the AI of these functions using Proposition 2 Item 3: $\mathsf{AI}(\mathsf{T}_{2^m,2^m+2k}) = \min(2^m, 2^m+2k-2^m+1) = 2k+1$ and $\mathsf{AI}(\mathsf{T}_{2^m,2^m+2k+1}) = \min(2^m, 2^m+2k+1-2^m+1) = 2k$ since $k < 2^{m-1}$.

Then, we prove the value of EAI for $f = \mathsf{T}_{2^m,2^m+2k}$. We begin by showing that $\mathsf{EAI}(f) \leq k$. Using Proposition 2 Item 4 we obtain $f = \sigma_{2^m,2^m+2k}$. The function

$g = \sigma_{k,2^m+2k} + \sigma_{2^m+k,2^m+2k}$ annihilates f: $fg = \sigma_{2^m+k,2^m+2k} + \sigma_{2^m+k,2^m+2k} = 0$ using Proposition 2 Item 2. Since g belongs to $\mathcal{F}_{k,2^m+2k-k}$, it gives $\mathsf{EAI}(f) \leq k$. To prove the other part, $\mathsf{EAI}(f) > k-1$, for any function h non null of degree d lower than k, we obtain that the product hf has degree lower than $2^m + k$ and is not null since $\mathsf{AI}(f) = 2k+1$ (note that the same arguments apply with $1+f$). The product of f with any function with monomials of degree between $2^m + 2k - d$ and $2^m + 2k$ is null or with monomials of degree greater than or equal to $2^m + 2k - d > 2^m + k$. Thereafter, no element of $\mathcal{F}_{d,n-d}$ annihilates f (nor $f+1$), therefore $\mathsf{EAI}(f) > k-1$, allowing to conclude $\mathsf{EAI}(f) = k$.

Finally, we prove the value of EAI for $f = \mathsf{T}_{2^m,2^m+2k}$. Using similar arguments as above for $\mathsf{T}_{2^m,2^m+2k}$, we can exhibit annihilators of f inside $\mathcal{F}_{(k+1),n-(k+1)}$ such as $\sigma_{k,2^m+2k+1} + \sigma_{2^m+k,2^m+2k+1}$ and $\sigma_{k+1,2^m+2k+1} + \sigma_{2^m+k+1,2^m+2k+1}$. Moreover, there are no annihilators in $\mathcal{F}_{k,n-k}$, since functions of degree at most k give non null products of degree at most $2^m + k$ and product with functions with monomials in the range of degree $[2^m + k + 1, 2^m + 2k + 1]$ give products null of with monomials of degree at least $2^m + 2k + 1 - k = 2^m + k + 1$. It allows to conclude, $\mathsf{EAI}(f) = k + 1$. □

Note that for such functions the EAI is (around) twice lower than the algebraic immunity. In particular, some functions from this family are example where the EAA is the most significant, the functions $\mathsf{T}_{2^m,2^{m+1}-1}$ have optimal AI, *i.e.*, $(n+1)/2$, but their EAI is only $n/4$.

5 Upper bound on EAI and functions such that $\mathsf{EAI} = \mathsf{AI}$

In this part we study upper bounds on the EAI and exhibit cases where the EAI is guaranteed to be not better than the AI. First, we show that the EAI is greater than the AI restricted to the slices of low Hamming weight. Then, we generalize a result of [CMR17] related to the algebraic immunity restricted to one slice. We prove in Theorem 2 that for functions obtained by direct sum, the restricted AI can be upper bounded by the AI of one component function minus the degree of the other component function. Finally, we use these results to exhibit cases where the EAI is at least AI plus one, or equal to AI.

Proposition 6. *Let $n \in \mathbb{N}^*$, and f an n-variable Boolean function, then:*

$$\forall k \in \mathbb{N}^*,\, k < \frac{n}{2}, \quad \mathsf{EAI}(f) \geq \mathsf{AI}_{\mathsf{P}_{0,k,n}}(f).$$

Proof. We fix $\mathsf{AI}_{\mathsf{P}_{0,k,n}}(f) = t$, using Definition 6 it means that f or $f+1$ admits an annihilator of degree t over $\mathsf{P}_{0,k,n}$ which is not null over $\mathsf{P}_{0,k,n}$, and this property does not hold for integers lower than t. Note that in the particular case of the set $\mathsf{P}_{0,k,n}$, the sub-matrix of $\mathbf{M}_{k,n}$ obtained by taking the columns corresponding to $\mathsf{P}_{0,k,n}$ is upper triangular with ones on the diagonal, then invertible, therefore there are no not null function of degree at most k null over $\mathsf{P}_{0,k,n}$. Accordingly, using the matrix representation, $\mathsf{AI}_{\mathsf{P}_{0,k,n}}(f) = t$ implies that:

$$\mathsf{rank}(\mathbf{M}_{t-1,n}(\mathsf{supp}_f \cap \mathsf{P}_{0,k,n})) = \mathsf{rank}(\mathbf{M}_{t-1,n}(\mathsf{supp}_f \cap \mathsf{P}_{0,k,n})) = \sum_{i=0}^{t-1} \binom{n}{i} = \mathsf{D}^n_{t-1}.$$

Since the rank of the matrix $\mathbf{M}_{t-1,n}(\mathsf{supp}_f \cap \mathsf{P}_{0,k,n})$ is already the maximal and it gives the rank of $\mathbf{M}_{t-1,n}(\mathsf{supp}_f)$, and the same argument on $\mathbf{M}_{t-1,n}(\mathsf{supp}_{f+1} \cap \mathsf{P}_{0,k,n})$ leads to $\mathsf{rank}(\mathbf{M}_{t-1,n}(\mathsf{supp}_{f+1})) = \mathsf{D}^n_{t-1}$.

We denote by $\mathbf{S} = \mathbf{M}_{n,n}(\mathsf{supp}_f)$, $\mathbf{C} = \mathbf{M}_{n,n}(\mathsf{supp}_{f+1})$ and $\mathbf{S}_i^j$ (respectively $\mathbf{C}_i^j$) the sub-matrix of $\mathbf{S}$ (respectively $\mathbf{C}$) formed by the rows indexed by the monomials from degree i to j. Then, in the previous paragraph we showed $\mathsf{rank}(\mathbf{S}_0^{t-1}) = \mathsf{rank}(\mathbf{C}_0^{t-1}) =$

D^n_{t-1} where the rank comes from the $\mathsf{P}_{0,k,n}$ part, and in the following we show than $\mathsf{span}(\mathbf{S}^n_{n-k}) \cap \mathsf{span}(\mathbf{S}^{t-1}_0) = 0_{2^n} = \mathsf{span}(\mathbf{C}^n_{n-k}) \cap \mathsf{span}(\mathbf{C}^{t-1}_0)$. By construction $\mathbf{S}^n_{n-k}, \mathbf{C}^n_{n-k}$ is a reordering of the last D^k_0 rows of $\mathbf{M}_{n,n}$ which corresponds to the monomials of degree at least $n-k$, therefore being null on all elements of Hamming weight lower than $n-k$, *a fortiori* on $\mathsf{P}_{0,k,n}$ since $k < n/2$. Then, all elements in the span of $\mathbf{S}^n_{n-k}$ (respectively $\mathbf{C}^n_{n-k}$) are null on $\mathsf{supp}_f \cap \mathsf{P}_{0,k,n}$ (respectively$\mathsf{supp}_{f+1} \cap \mathsf{P}_{0,k,n}$) whereas only the null vector has this property in the span of $\mathbf{S}^{t-1}_0$ (respectively $\mathbf{C}^{t-1}_0$).

Finally, since $\mathbf{S}^n_{n-t-1}$ is a sub-matrix of $\mathbf{S}^n_{n-k}$, we obtain $\mathsf{rank}(\mathbf{S}^{t-1}_0 | \mathbf{S}^n_{n-t-1}) = \mathsf{D}^n_{t-1} + \mathsf{rank}(\mathbf{S}_{n-t-1,n})$, and the same result relatively to $\mathbf{C}$. Therefore, using Proposition 3 we can conclude $\mathsf{EAI}(f) \geq t$ hence $\mathsf{EAI}(f) \geq \mathsf{AI}_{\mathsf{P}_{0,k,n}}(f)$.

□

In the case of functions obtained by direct sum, Theorem 1 of [CMR17] gives an upper bound on the algebraic immunity restricted to a slice depending on the (standard) algebraic immunity of one of the two functions and the degree of the second one. We generalize this result, it allows to derive an upper bound on the AI restricted to $\mathsf{P}_{0,d,n+m}$ of $f+g$ depending on the AI of f and the degree of g. Combining it with the bound of Proposition 6, it gives an upper bound on the EAI of a direct sum, and it allows to determine functions such that the AI and EAI have the same value.

Theorem 2. *Let $n, m \in \mathbb{N}$, and $S \subseteq \mathbb{F}_2^{n+m}$, if for all elements $(a,b) \in S$ with $a \in \mathbb{F}_2^n$ and $b \in \mathbb{F}_2^m$ there exists a vectorial Boolean function $L : \mathbb{F}_2^n \to \mathbb{F}_2^m$ satisfying the following properties:*

- *the m coordinate functions of L are affine,*
- *$L(a) = b$,*
- *$\forall x \in \mathbb{F}_2^n$ $(x, L(x)) \in S$,*

then for all functions $f \in \mathcal{B}_n$, $g \in \mathcal{B}_m$ and their direct sum ψ the following holds:

$$\mathsf{AI}_S(\psi) \geq \mathsf{AI}(f) - \mathsf{deg}(g).$$

Proof. Let $h(x,y)$ be a non-null annihilator of ψ over S of degree $\mathsf{AI}_S(\psi)$), then there exists $(a,b) \in S$ such that $h(a,b) = 1$. Assuming the existence of L satisfying the three requirements, $h(x, L(x))$ is an n-variable Boolean function annihilator of $f(x) + g(L(x))$ since $(x, L(x)) \in S$ and h annihilates ψ over S. Moreover, the function $h(x, L(x))$ is not null since $h(a, L(a)) = h(a,b) = 1$.

If $g(b) = 0$ then $h(x, L(x))(1 + g(L(x))$ is a non null annihilator of f, which gives:

$$\mathsf{AN}(f) \leq \mathsf{deg}(h) + \mathsf{deg}(L(g)).$$

Since $h(x,y)$ is a non null annihilator of ψ over S we get $\mathsf{deg}(h(x, L(x))) \leq \mathsf{AI}_S(\psi)$, and since all coordinate functions of L are affine $\mathsf{deg}(L(g)) \leq \mathsf{deg}(g)$, it implies $\mathsf{AN}(f) \leq \mathsf{AI}_S(\psi) + \mathsf{deg}(g)$.

If $g(b) = 1$, then $h(x, L(x))g(L(x)$ is a non null annihilator of $1 + f$, which gives $\mathsf{AN}(f) \leq \mathsf{deg}(h) + \mathsf{deg}(L(g))$. Following the same arguments, $\mathsf{AN}(f+1) \leq \mathsf{AI}_S(\psi) + \mathsf{deg}(g)$, and combining the two cases for $g(b)$ we obtain: $\mathsf{AI}(f) \leq \mathsf{AI}_S(\psi) + \mathsf{deg}(g)$.

This result has been derived assuming $h(x,y)$ be a non-null annihilator of ψ over S of degree $\mathsf{AI}_S(\psi)$, when it is not the case it implies the existence of $h(x,y)$ a non-null annihilator of $1+\psi$ over S of degree $\mathsf{AI}_S(\psi)$ by definition of AI_S. Accordingly, the same reasoning applies with $1+\psi$, $f+1$ and g, therefore we can conclude $\mathsf{AI}(f) \leq \mathsf{AI}_S(\psi) + \mathsf{deg}(g)$ or equivalently:

$$\mathsf{AI}_S(\psi) \geq \mathsf{AI}(f) - \mathsf{deg}(g).$$

□

In particular the result of [CMR17] consists in the case where $n \leq k \leq m$ and $S = \mathsf{E}_{k,n+m}$. Up to permutations of the variables, L is chosen to give the complement of a on the first n bits, 1 on $k-n$ remaining bits and 0 on the others. Such L maps a to b, all elements of $\mathbb{F}_2^n$ to elements of Hamming weight $\mathsf{w_H}(a)+n-\mathsf{w_H}(a)+k-n=k$ and each coordinate function is affine.

Corollary 1. *Let $k,n,m \in \mathbb{N}$, $n \leq m$, $n \leq k \leq n+m$ and $S = \mathsf{P}_{0,k,n+m}$, for all functions $f \in \mathcal{B}_n$, $g \in \mathcal{B}_m$ and their direct sum ψ the following holds:*

$$\mathsf{AI}_S(\psi) \geq \mathsf{AI}(f) - \mathsf{deg}(g).$$

Proof. Using Theorem 2, it is sufficient to show for $(a,b) \in S$ the existence of L satisfying the requirements. Up to permutations of the variables, the $\mathsf{w_H}(a) = r$ first bits of a are equal to one and the $n-r$ others to zero, the $\mathsf{w_H}(a,b) - r = s$ first bits of b are equal to one and the $m-s$ others to zero.

If $s \leq n-r$ we define L as $L(x_1,\ldots,x_n) = (1+x_{r+1},\ldots,1+x_{r+s},0_{m-s})$. It satisfies $L(a)=b$, for all element $x \in \mathbb{F}_2$ the vector $(x, L(x))$ has Hamming weight at most n then $(x,L(x)) \in S$, and all coordinate functions of L are affine, therefore L complies with the requirements of the theorem.

If $s > n-r$ we define L as $L(x_1,\ldots,x_n) = (1+x_{r+1},\ldots,1+x_n,1_{s-n+r},0_{m-s})$. It satisfies $L(a)=b$, for all element $x \in \mathbb{F}_2$ the vector $(x,L(x))$ has Hamming weight at most $n+s-n+r = s+r = \mathsf{w_H}((a,b))$ then $(x,L(x)) \in S$ since all elements of Hamming $\mathsf{w_H}((a,b))$ belong to S, and all coordinate functions of L are affine, therefore L complies with the requirements of the theorem.

□

Proposition 7. *Let $n,m \in \mathbb{N}^*$, $m > n$, and f an n-variable Boolean functions. We denote by $\psi \in \mathcal{B}_{n+m}$ the function defined for all (x,y) with $x \in \mathbb{F}_2^n$, $y \in \mathbb{F}_2^m$ as $\psi(x,y) = f(x)$. The following holds:*

$$\mathsf{AI}(\psi) = \mathsf{EAI}(\psi).$$

Proof. First we apply Corollary 1 with f, the null function in m variables as g and $S = \mathsf{P}_{0,n,n+m}$, it gives:

$$\mathsf{AI}_{\mathsf{P}_{0,n,n+m}}(\psi) \geq \mathsf{AI}(f).$$

Then, using Proposition 6 for $k=n$ we obtain:

$$\mathsf{EAI}(\psi) \geq \mathsf{AI}_{\mathsf{P}_{0,n,n+m}}(\psi).$$

Since the algebraic immunity is an affine equivalent notion $\mathsf{AI}(\psi) = \mathsf{AI}(f)$ therefore $\mathsf{EAI}(\psi) \geq \mathsf{AI}(\psi)$. Finally, by definition of the EAI, $\mathsf{EAI}(\psi) \leq \mathsf{AI}(\psi)$, which allows us to conclude:

$$\mathsf{EAI}(\psi) = \mathsf{AI}(\psi).$$

□

From Corollary 1 we can deduce that all functions that are direct sums of an n-variable Boolean function f and the sum of m variables such that $n \leq m$ are such that $\mathsf{EAI}(f) \geq \mathsf{AI}(f) - 1$. The Xor-Threshold functions used to instantiate FiLIP [MCJS19b, HMR20] belong to this category, therefore the extremal algebraic attack would lead to no improvement or low improvement of the algebraic attack on these specific functions.

Proposition 7 shows that for all functions with more than half variables with no influence (variables such that changing their values never changes the output), the EAI and the AI are equal. There are examples of functions using less than half of the (key/seed) variables with all instances of the cipher FiLIP, and with the local pseudorandom generator of Goldreich [Gol00].

Note that these results can be generalized to other variants of the algebraic immunity than the EAI. Indeed, Proposition 6 bounds the EAI based on the algebraic properties of the function only on the set formed by the slices of small Hamming weight, then the same arguments apply when we consider other set of monomials including the same slices. For example, the reasoning applies if we consider the set given by $\mathsf{P}_{1,d,n}$ and only a subpart of $\mathsf{P}_{n-d,n,n}$. Similarly, the generality of Theorem 2 can be used to derive results on variations of the AI for direct sums, as in Corollary 1 for case of EAI.

6 Applications of EAA for functions in the literature

In this section, we investigate potential applications of the EAA. Firstly, following the proof strategy for Theorem 1, we determine the EAI of two filter functions used in ciphers GEA-1/2 and LILI-128 by implementing Proposition 3 to compute the rank of sub-matrices of the punctured Reed-Muller code, and get the result when the equality does not hold. More specifically, in Subsection 6.1 for the filter function of GEA-1/2, and in Subsection 6.2 for the filter function of LILI-128, we respectively compute the exact value of the EAI, and find all the corresponding annihilators and the linearly independent ones among them which might be used for further attacks. We note the existence of attacks targeting the initialization and/or the non-linear filter function of GEA-1/2, and the non-linear filter function of LILI-128. Here, we use their non-linear filter functions as examples to illustrate the computation of the EAI. Next, in Subsection 6.3, we review existing symmetric primitives (such as FLIP, FiLIP and variants) that triggered EAA, and we explain the detailed reason why EAA cannot apply directly.

6.1 GEA-1 and GEA-2

GPRS (General Packet Radio Service) is a mobile data standard that was widely deployed in the early 2000s. To protect against eavesdropping GPRS between the phone and the base station, two proprietary stream ciphers GEA-1 and GEA-2 were initially designed and used for this purpose. GEA-1 is built from three linear feedback shift registers over $\mathbb{F}_2$, together with a non-linear filter function $f : \mathbb{F}_2^7 \to \mathbb{F}_2$, which is a Boolean function on seven variables of degree 4. The first public analysis of GEA-1 was proposed in [BDL+21] as a key recovery attack utilizing the weakness of the initialization function. Without such a weakness in the initialization as in GEA-1, the authors also presented key recovery attacks on GEA-2. The attacks on GEA-1/2 were further improved / complemented by Amzaleg and Dinur [AD22] and Ding, Wu, Wang, Guan and Li [DWW+22].

As said, our focus is not on the initialization function, but only on the component of the key generation function f. We take the specification of $f = f(x_1, x_2, \ldots, x_7)$ from [BDL+21] and give it in algebraic normal form as follows:

$$\begin{aligned}&x_1x_3x_6x_7 + x_1x_4x_6x_7 + x_1x_2x_6x_7 + x_2x_3x_6x_7 + x_1x_3x_4x_7 + x_2x_4x_5x_7 + x_2x_4x_6x_7+\\&x_1x_3x_5 + x_1x_3x_4 + x_1x_2x_4 + x_1x_3x_7 + x_1x_2x_5 + x_1x_2x_7 + x_2x_3x_7 + x_3x_6x_7 + x_1x_4x_6+\\&x_2x_5x_7 + x_2x_3x_6 + x_1x_4 + x_1x_6 + x_2x_4 + x_2x_6 + x_2x_7 + x_1x_3 + x_2 + x_3x_4 + x_3x_6+\\&x_3x_7 + x_5x_6 + x_6x_7 + x_3 + x_4 + x_6.\end{aligned}$$

The GEA-2 cipher is a simple extension of GEA-1. A fourth register of length 29, is added to the system together with an instance of f. In this paper, we focus only on the filter function f, and it is the same for GEA-1 and GEA-2, so we can call them uniformly GEA.

According to the definitions of AI and EAI, for the filter function of GEA we have $\mathsf{AI}(f) = \mathsf{EAI}(f) = 3$. In addition, we found $\mathcal{C}\mathsf{EAN}(f) = 64$ usable annihilators[2], and 48

[2] We have discarded the annihilators with null part in the set of high degree $\mathcal{F}_{n-d}$ since they are not interesting for the attacks.

linearly independent ones. We give one example in the following:

$$x_1x_2x_4x_5x_6x_7 + x_1x_2x_3x_4x_5 + x_1x_2x_3x_5x_7 + x_1x_2x_4x_6x_7 + x_1x_3x_4x_6x_7 + x_1x_4x_6x_7+ \\ x_1x_2x_3 + x_1x_2x_4 + x_1x_3x_5 + x_1x_4x_5 + x_1x_2x_7 + x_1x_5x_7 + x_1x_2 + x_1x_3 + x_1x_4 + x_1x_5+ \\ x_1x_7 + x_1.$$

6.2 LILI-128

LILI-128 is a candidate stream cipher submitted to the NESSIE project by Simpson, Dawson, Golic and Millan [SDGM00]. It did not pass the first stage of the contest because the non-linear filter function was successfully reconstructed in [HHL+07]. It uses two LFSRs, LFSR$_c$ and LFSR$_d$. LFSRc has an internal state of 39 bits and is clocked once for each output bit. LFSR$_d$ has an internal state of 89 bits and is clocked 1 to 4 times, depending on two bits in LFSR$_c$. During key setup phase a $128 = 39+89$-bit cryptovariable is directly loaded into these two registers. If we use $u_0, u_1, \ldots, u_{88}$ to denote the individual bits of LFSR$_d$, then the ten bits from LFSR$_d$ are fed to a highly nonlinear function, $f_d : \mathbb{F}_2^{10} \to \mathbb{F}_2$ to generate one output bit $z(t)$ as

$$z(t) = f_d(u_0, u_1, u_3, u_7, u_{12}, u_{20}, u_{30}, u_{44}, u_{65}, u_{80}).$$

The ten-variable Boolean function $f_d(x_1, x_2, \cdots, x_{10})$ has the following ANF:

$$x_2 + x_3 + x_4 + x_5 + x_6x_7 + x_1x_8 + x_2x_8 + x_1x_9 + x_3x_9 + x_4x_{10} + x_6x_{10} + x_3x_7x_9+ \\ x_4x_7x_9 + x_6x_7x_9 + x_3x_8x_9 + x_6x_8x_9 + x_4x_7x_{10} + x_5x_7x_{10} + x_6x_7x_{10} + x_3x_8x_{10}+ \\ x_4x_8x_{10} + x_2x_9x_{10} + x_3x_9x_{10} + x_4x_9x_{10} + x_5x_9x_{10} + x_3x_7x_8x_{10} + x_5x_7x_8x_{10}+ \\ x_2x_7x_9x_{10} + x_4x_7x_9x_{10} + x_6x_7x_9x_{10} + x_1x_8x_9x_{10} + x_3x_8x_9x_{10} + x_4x_8x_9x_{10}+ \\ x_6x_8x_9x_{10} + x_4x_6x_7x_9 + x_5x_6x_7x_9 + x_2x_7x_8x_9 + x_4x_7x_8x_9 + x_4x_6x_7x_9x_{10}+ \\ x_5x_6x_7x_9x_{10} + x_3x_7x_8x_9x_{10} + x_4x_7x_8x_9x_{10} + x_4x_6x_7x_8x_9 + x_5x_6x_7x_8x_9+ \\ x_4x_6x_7x_8x_9x_{10} + x_5x_6x_7x_8x_9x_{10}.$$

According to definitions of AI and EAI, for the filter function of LILI-128 we obtain $\mathsf{AI}(f) = \mathsf{EAI}(f) = 4$. In addition, we found 264 annihilators, and $\mathcal{C}\mathsf{EAN}(f) = 151$ of them having degree up to 4 in the lower part set of $\mathcal{F}_d$. We give one example in the following:

$$\begin{aligned}
&x_1x_4x_5x_7x_8x_9x_{10} + x_2x_4x_5x_7x_8x_9x_{10} + x_3x_4x_5x_7x_8x_9x_{10} + x_1x_2x_6x_7x_8x_9x_{10}+\\
&x_1x_3x_6x_7x_8x_9x_{10} + x_4x_5x_6x_7x_8x_9x_{10} + x_5x_6x_7x_8x_9x_{10} + x_1x_3x_7x_8 + x_3x_4x_7x_8+\\
&x_1x_5x_7x_8 + x_4x_5x_7x_8 + x_1x_6x_7x_8 + x_2x_6x_7x_8 + x_1x_2x_7x_9 + x_1x_3x_7x_9 + x_2x_3x_7x_9+\\
&x_1x_5x_7x_9 + x_2x_5x_7x_9 + x_4x_5x_7x_9 + x_1x_2x_8x_9 + x_1x_3x_8x_9 + x_2x_3x_8x_9 + x_2x_4x_8x_9+\\
&x_4x_5x_8x_9 + x_1x_6x_8x_9 + x_4x_6x_8x_9 + x_1x_7x_8x_9 + x_2x_7x_8x_9 + x_4x_7x_8x_9 + x_2x_4x_7x_{10}+\\
&x_3x_4x_7x_{10} + x_1x_5x_7x_{10} + x_3x_5x_7x_{10} + x_1x_6x_7x_{10} + x_3x_6x_7x_{10} + x_4x_6x_7x_{10}+\\
&x_1x_2x_8x_{10} + x_2x_3x_8x_{10} + x_1x_4x_8x_{10} + x_3x_4x_8x_{10} + x_1x_5x_8x_{10} + x_2x_5x_8x_{10}+\\
&x_3x_5x_8x_{10} + x_1x_6x_8x_{10} + x_2x_7x_8x_{10} + x_3x_7x_8x_{10} + x_1x_2x_9x_{10} + x_1x_3x_9x_{10}+\\
&x_2x_3x_9x_{10} + x_1x_4x_9x_{10} + x_2x_4x_9x_{10} + x_2x_5x_9x_{10} + x_3x_5x_9x_{10} + x_1x_6x_9x_{10}+\\
&x_3x_6x_9x_{10} + x_1x_7x_9x_{10} + x_1x_8x_9x_{10} + x_2x_8x_9x_{10} + x_3x_8x_9x_{10} + x_1x_2x_7 + x_1x_3x_7+\\
&x_2x_3x_7 + x_1x_4x_7 + x_2x_4x_7 + x_1x_5x_7 + x_2x_5x_7 + x_4x_6x_7 + x_5x_6x_7 + x_1x_2x_8 + x_2x_3x_8+\\
&x_3x_4x_8 + x_1x_5x_8 + x_4x_5x_8 + x_1x_7x_8 + x_3x_7x_8 + x_4x_7x_8 + x_5x_7x_8 + x_1x_2x_9 + x_1x_3x_9+\\
&x_2x_3x_9 + x_2x_4x_9 + x_3x_4x_9 + x_1x_5x_9 + x_3x_5x_9 + x_4x_5x_9 + x_4x_7x_9 + x_2x_8x_9 + x_3x_8x_9+\\
&x_4x_8x_9 + x_5x_8x_9 + x_1x_2x_{10} + x_1x_3x_{10} + x_2x_3x_{10} + x_1x_4x_{10} + x_3x_4x_{10} + x_1x_5x_{10}+\\
&x_2x_5x_{10} + x_5x_6x_{10} + x_2x_7x_{10} + x_3x_7x_{10} + x_1x_8x_{10} + x_4x_8x_{10} + x_6x_8x_{10} + x_7x_8x_{10}+\\
&x_1x_9x_{10} + x_4x_9x_{10} + x_5x_9x_{10} + x_6x_9x_{10} + x_8x_9x_{10} + x_1x_2 + x_1x_3 + x_2x_3 + x_1x_4+\\
&x_3x_4 + x_1x_5 + x_3x_5 + x_1x_7 + x_2x_7 + x_3x_7 + x_4x_7 + x_5x_7 + x_1x_8 + x_2x_8 + x_3x_8 + x_4x_8+\\
&x_5x_8 + x_7x_8 + x_1x_9 + x_2x_9 + x_3x_9 + x_4x_9 + x_5x_9 + x_8x_9 + x_1x_{10} + x_4x_{10} + x_6x_{10}+\\
&x_7x_{10} + x_1 + x_2 + x_3 + x_4 + x_5 + x_7 + x_8 + x_9 + 1.
\end{aligned}$$

6.3 Application Scope and limitations of EAA

In this part we discuss the scope of the extremal algebraic attack, that is, symmetric primitives where other subsets of monomials than the ones of low degree can be kept stable by the updating process. We explain why a direct application of the extremal algebraic attack is not possible for these already published designs, and suggest attack modifications. First we recall the paradigms of the stream ciphers FLIP [MJSC16] and FiLIP [MCJS19b] since we consider (modifications of) these schemes.

The Filter Permutator (FP) paradigm is a stream cipher paradigm introduced in [MJSC16] in the context of hybrid homomorphic encryption [NLV11], designed to be efficiently evaluated homomorphically. The filter permutator paradigm is depicted in Figure 1 on the left side. For each bit of keystream the binary key is permuted by a wire-cross permutation publicly derived from a pseudorandom generator and then a Boolean function called filter is applied on this permuted key to give the keystream bit. The improved filter permutator, introduced in [MCJS19b] modifies the FP paradigm by using only a subpart of the key for each keystream bit and adding a random vector to the input of the filter function.

The attack based on the extremal algebraic immunity adapts differently to FLIP, FiLIP and variations of these schemes, as we detail in the following:

FLIP. For FLIP stream ciphers, the inputs of the Boolean function f are always the variables x_i of the secret key K, only permuted by a wire-cross permutation. Accordingly, the product $f \cdot g$ with $g \in \mathcal{F}_{d,n-d}$ an annihilator of f gives equations with monomials in the variables x_i in $\mathcal{F}_{d,n-d}$, and the wire-cross permutations stabilize the set $\mathsf{P}_{0,d,n} \cup \mathsf{P}_{n-d,n,n}$. Instead of the usual algebraic attack, the extremal algebraic attack can directly be used. It is the only context we found in open literature where the attack applies.

Nevertheless, the EAA does not give an attack with better complexity than already known, nor contradict the 2^{128} security claim for two reasons. First, the instances of FLIP (called FLIP functions in [MJSC16]) are direct sums of monomials, they correspond to functions with AI far from the maximum of $\lceil n/2 \rceil$. In this case the AI of the function comes only from the part of low degree (the ANF contains only elements of low degree), thereafter the EAI can be bounded from the AI restricted on the slices of low Hamming weight, following Proposition 6. It results in cases where $\mathsf{AI}(f) = \mathsf{EAI}(f)$, where the EAA has a worse time complexity.

Then, since the Hamming weight is constant and known for the keys of FLIP instances, the attacks using the properties of the filter function on the particular slice of Hamming weight $n/2$ from [CMR17] are more adapted.

Note that the extremal algebraic attack can apply to variants of FLIP with different filters. For example, if the filter is a function XOR_k in direct sum with the threshold function $\mathsf{T}_{2^m,2^{m+1}-1}$ (an example from Proposition 5), the resulting function has algebraic immunity equal to 2^m. Its nonlinearity can be derived from the formula for the nonlinearity of xor-threshold functions ([CM22], Proposition 7), which is sufficient to resist correlation-based attacks in FLIP, provided the parameters are large enough. Therefore, the function would be considered secure in that context. However, its EAI is only $2^{m-1} + k + 1$ hence the gap between the EAI and the AI indicates that the function would be secure against algebraic attacks but not against EAA.

FLIP with whitening. We consider a variant of FLIP where a whitening is added before the application of the filter, this alternative would be sufficient to avoid the filter to be evaluated on inputs of Hamming weight $n/2$ only. We explore two possible strategies to apply the EAA.

First, we consider the n key bits and their complements as $2n$ binary variables, in this case the adversary obtains a system in $2n$ variables. This choice is motivated by the fact that $\mathsf{P}_{n-d,n,n}$ is not stable when constants are added: the affine mapping $x_i \mapsto x_i + 1$ can generate monomials of lower degree. But as for FLIP, the attack generalization leads to improvements only for filter functions such that the EAI would be different from the AI. If we write the filter function as a $2n$-variable function in the $2n$ key variables, it corresponds to a direct sum of the initial filter and the null function in n variables, therefore a function such that the EAI equals the AI by Proposition 7.

The other strategy consists in considering only the n original variables, in this case a variation of the EAA is possible. Each time the Hamming weight of the whitening is at most t, an annihilator from $\mathcal{F}_{d,n-d}$ gives equations with monomials belonging to $\mathsf{P}_{0,d,n} \cup \mathsf{P}_{n-d-t,n,n}$. Then, it is interesting for functions having annihilators with a part of degree at most d and potentially a part of degree even higher than $n - d$, such that subtracting t to the degree does not go lower than $n - d$ (which would result in more monomials). Indeed, on a monomial of degree d, the mapping $x \mapsto x + a$ with $a \in \mathbb{F}_2^n$ of Hamming weight $t \leq d$ can generate monomials of any degree between $d - t$ and d. For example on the monomial $\prod_{i=1}^{d} x_i$, the mapping $x \mapsto x + 1_d$ gives the sum of all monomials of degree between 0 and d in the variables x_1 to x_d.

FLIP with a large register, and local PseudoRandom Generators (PRG). We consider an alternative of FLIP where the key register is larger than the number of variables of the filter function. This setting also corresponds to the local variant of Goldreich's PRG [Gol00] where the seed's size is a parameter n and the number of variables of the function (called predicate) is a constant. We refer to the survey of Applebaum [App13] for local PRGs and to [AL16, CDM+18, YGJL22, Üna23] for recent cryptanalyses.

In this context f has n variables but the register has size $N > n$, that is, the output is independent of a large number of variables. It is also the case of Goldreich's PRG with a constant locality, where $N \gg n$. The EAA applies to this context, nevertheless the filter function corresponds to the direct sum of f in n variables and the null function in

$N-n$ variables, which is the case of Corollary 1, so the attack does not lead to a better complexity than the standard algebraic attack.

We can conceive a variant of Goldreich's PRG with an anti-local property (which would go against the motivation of the first design and the following lines of works), where each output bit depends on all or almost all inputs. In this case, the EAI will be a criterion to consider for the security, since it gives a better attack than the one based on the AI, and predicates satisfying the requirements of [AL16] will not be immune to the extremal algebraic attack. Nevertheless, we are not aware of contexts where such anti-local PRG would be interesting.

FiLIP. FiLIP uses both a large key register and a whitening, which limits the impact of EAA as explained above. Furthermore, the different filters considered so far are direct sum of monomials [MCJS19b] and functions obtained as the direct sum of a linear function and a threshold function [MCJS19a, HMR20]. As for FLIP instances, the first family of function is such that $\mathsf{AI}=\mathsf{EAI}$, and for the second family, the direct sum with a linear function corresponds to a case covered by Corollary 1 resulting in a difference between EAI and AI of at most 1.

Beyond variants of FLIP and FiLIP, we discuss variations of the extremal algebraic attack that could lead to new cryptanalyses on filtered linear feedback shift registers or nonlinear feedback shift registers.

Adaptation to filtered LFSR. In the context of a filtered LFSR, due to the linear update of the variables, the monomials of degree at most d stay in $\mathsf{P}_{0,d,n}$, but the high degree part is not stable. Each affine mapping $x_i \mapsto \varepsilon + \sum_{j\in J} x_i$ can give monomials of lower degree. An attack strategy consists in selecting only the keystream bits such that the associated linear updates are only a permutation of the variables of the initial state, in this case an annihilator from $\mathcal{F}_{d,n-d}$ gives equations only in monomials from $\mathsf{P}_{0,d,n}$ and $\mathsf{P}_{n-d-,n,n}$. These cases being extremely rare (Over the $\prod_{i=0}^{n-1}(2^n-2^i)$ possible non-degenerate linear mappings, only $n!$ correspond to a permutation of the initial n variables), therefore the adversary should also take into consideration the cases where the linear update does not reduce too much the degree of the monomials from the $\mathsf{P}_{n-d,n,n}$ part. Considering n linear (not affine) mappings, the degree can degrade at most from the maximum occurrence of one variable, that we denote by ℓ, in this case, the monomials created belong to $\mathsf{P}_{n-d-\ell,n,n}$. For example, the attack could be interesting for a function having $\mathsf{AI}(f)>d$ and an annihilator in $\mathcal{F}_{d,n-d}$ where only the monomial of degree n appears in the $\mathsf{P}_{n-d,n,n}$ part.

Adaptation to filtered Nonlinear Feedback Shift Register (NFSR). In this context, the update is not linear so the degree of the equations increases, and the same happens for the monomials of the annihilators. A variant of the attack could be over the monomials of high degree only, since the degree increases quickly, and the one of the $\mathsf{P}_{n-d,n,n}$ part decreases less. With a preliminary evaluation, the attack does not appear to achieve better complexity. A particular study could be performed to verify if the low degree monomials disappear in specific cases.

7 Conclusion

In this article we propose the new notion of extremal algebraic immunity, to illustrate and study potential generalizations of the algebraic attack presented by Courtois and Meier's twenty years ago. We perform a theoretic study of the EAI criterion and explore its relation to other algebraic criteria. This algebraic attack does not give a better complexity than Courtois and Meier's attacks on the public stream ciphers, but it can help to understand better the strength of the standard algebraic attack and avoid weaknesses in the construction of future stream cipher designs.

As for future works, it might be interesting to determine if variations of the EAA can be applied to new stream ciphers adapted to advanced applications such as fully homomorphic encryption, multiparty computation or zero knowledge. Another direction we can investigate is probabilistic EAA. Similarly to probabilistic AA mentioned in related work, the high level idea is to find functions annihilating the filter function in most inputs but not all, this degree of freedom could give a bigger number of exploitable equations in some cases.

Acknowledgment.

Pierrick Méaux was funded by the European Research Council (ERC) under the Advanced Grant program (reference number: 787390). Qingju Wang is supported by ANR through the ANR-23-CE39-0009 TRUST project. The authors thank Lin Jiao and Yonglin Hao for checking the calculation of EAI for the filter functions in this article, and they thank Willi Meier for his feedback on an earlier version of this article.

References

[AD22] Dor Amzaleg and Itai Dinur. Refined cryptanalysis of the GPRS ciphers GEA-1 and GEA-2. In Orr Dunkelman and Stefan Dziembowski, editors, *EUROCRYPT 2022, Part III*, volume 13277 of *LNCS*, pages 57–85. Springer, 2022. doi:10.1007/978-3-031-07082-2_3.

[AL16] Benny Applebaum and Shachar Lovett. Algebraic attacks against random local functions and their countermeasures. In Daniel Wichs and Yishay Mansour, editors, *STOC 2016*, pages 1087–1100. ACM, 2016. doi:10.1145/2897518.2897554.

[App13] Benny Applebaum. Cryptographic hardness of random local functions-survey. In Amit Sahai, editor, *TCC 2013*, volume 7785 of *LNCS*, page 599. Springer, 2013. doi:10.1007/978-3-642-36594-2_33.

[BDL+21] Christof Beierle, Patrick Derbez, Gregor Leander, Gaëtan Leurent, Håvard Raddum, Yann Rotella, David Rupprecht, and Lukas Stennes. Cryptanalysis of the GPRS encryption algorithms GEA-1 and GEA-2. In Anne Canteaut and François-Xavier Standaert, editors, *EUROCRYPT 2021, Part II*, volume 12697 of *LNCS*, pages 155–183. Springer, 2021. doi:10.1007/978-3-030-77886-6_6.

[BP05a] An Braeken and Bart Preneel. On the algebraic immunity of symmetric Boolean functions. In *INDOCRYPT 2005*, volume 3797 of *LNCS*, pages 35–48. Springer, 2005. doi:10.1007/11596219_4.

[BP05b] An Braeken and Bart Preneel. Probabilistic algebraic attacks. In Nigel P. Smart, editor, *Cryptography and Coding*, pages 290–303, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg. doi:10.1007/11586821_20.

[Car21] Claude Carlet. *Boolean Functions for Cryptography and Coding Theory*. Cambridge University Press, 2021. doi:10.1017/9781108606806.

[CDM+18] Geoffroy Couteau, Aurélien Dupin, Pierrick Méaux, Mélissa Rossi, and Yann Rotella. On the Concrete Security of Goldreich's Pseudorandom Generator. In Thomas Peyrin and Steven D. Galbraith, editors, *ASIACRYPT 2018, Part II*, volume 11273 of *LNCS*, pages 96–124. Springer, 2018. doi:10.1007/978-3-030-03329-3_4.

[CM03] Nicolas T. Courtois and Willi Meier. Algebraic attacks on stream ciphers with linear feedback. In Eli Biham, editor, *EUROCRYPT 2003*, volume 2656 of *LNCS*, pages 345–359. Springer, 2003. doi:10.1007/3-540-39200-9_21.

[CM22] Claude Carlet and Pierrick Méaux. A complete study of two classes of Boolean functions: Direct sums of monomials and threshold functions. *IEEE Trans. Inf. Theory*, 68(5):3404–3425, 2022. doi:10.1109/TIT.2021.3139804.

[CMR17] Claude Carlet, Pierrick Méaux, and Yann Rotella. Boolean functions with restricted input and their robustness; application to the FLIP cipher. *IACR Trans. Symmetric Cryptol.*, 2017(3), 2017. doi:10.13154/TOSC.V2017.I3.192-227.

[Cou02] Nicolas T. Courtois. Higher order correlation attacks, XL algorithm and cryptanalysis of toyocrypt. In Pil Joong Lee and Chae Hoon Lim, editors, *Information Security and Cryptology - ICISC 2002*, volume 2587 of *LNCS*, pages 182–199. Springer, 2002. doi:10.1007/3-540-36552-4_13.

[Cou03] Nicolas T. Courtois. Fast algebraic attacks on stream ciphers with linear feedback. In Dan Boneh, editor, *CRYPTO 2003*, volume 2729 of *LNCS*, pages 176–194. Springer, 2003. doi:10.1007/978-3-540-45146-4_11.

[Did06] Frédéric Didier. A new upper bound on the block error probability after decoding over the erasure channel. *IEEE Transactions on Information Theory*, 52(10):4496–4503, 2006. doi:10.1109/TIT.2006.881719.

[DWW$^+$22] Lin Ding, Zheng Wu, Xinhai Wang, Ziyu Guan, and Mingjin Li. New attacks on the GPRS encryption algorithms GEA-1 and GEA-2. *IEEE Trans. Inf. Forensics Secur.*, 17:2878–2889, 2022. doi:10.1109/TIFS.2022.3197064.

[Fau99] Jean-Charles Faugère. A new efficient algorithm for computing Groebner bases. *Journal of Pure and Applied Algebra*, 139:61–88, june 1999. doi:10.1016/S0022-4049(99)00005-5.

[Fau02] Jean-Charles Faugère. A new efficient algorithm for computing Grobner bases without reduction to zero. In *Workshop on application of Groebner Bases 2002*, Catania, Spain, 2002. doi:hal.inria.fr/inria-00100997.

[Gol00] Oded Goldreich. Candidate one-way functions based on expander graphs. *Electronic Colloquium on Computational Complexity (ECCC)*, 7(90), 2000. https://eccc.weizmann.ac.il/eccc-reports/2000/TR00-090/index.html,.

[HHL$^+$07] Xiangao Huang, Wei Huang, Xiaozhou Liu, Chao Wang, Zhu jing Wang, and Tao Wang. Reconstructing the nonlinear filter function of LILI-128 stream cipher based on complexity, 2007. arXiv:cs/0702128.

[HMR20] Clément Hoffmann, Pierrick Méaux, and Thomas Ricosset. Transciphering, using filip and TFHE for an efficient delegation of computation. In Karthikeyan Bhargavan, abeth Oswald, and Manoj Prabhakaran, editors, *INDOCRYPT 2020*, volume 12578 of *LNCS*, pages 39–61. Springer, 2020. doi:10.1007/978-3-030-65277-7_3.

[MCJS19a] Pierrick Méaux, Claude Carlet, Anthony Journault, and François-Xavier Standaert. Improved filter permutators: Combining symmetric encryption design, Boolean functions, low complexity cryptography, and homomorphic encryption, for private delegation of computations. Cryptology ePrint Archive, Report 2019/483, 2019. https://eprint.iacr.org/2019/483.

[MCJS19b] Pierrick Méaux, Claude Carlet, Anthony Journault, and François-Xavier Standaert. Improved filter permutators for efficient FHE: better instances and implementations. In Feng Hao, Sushmita Ruj, and Sourav Sen Gupta, editors, *INDOCRYPT*, volume 11898 of *LNCS*, pages 68–91. Springer, 2019. doi:10.1007/978-3-030-35423-7_4.

[Méa19] Pierrick Méaux. On the fast algebraic immunity of majority functions. In Peter Schwabe and Nicolas Thériault, editors, *LATINCRYPT*, volume 11774 of *LNCS*, pages 86–105. Springer, 2019. doi:10.1007/978-3-030-30530-7_5.

[MJSC16] Pierrick Méaux, Anthony Journault, François-Xavier Standaert, and Claude Carlet. Towards stream ciphers for efficient FHE with low-noise ciphertexts. In Marc Fischlin and Jean-Sébastien Coron, editors, *EUROCRYPT 2016, Part I*, volume 9665 of *LNCS*, pages 311–343. Springer, 2016. doi:10.1007/978-3-662-49890-3_13.

[MPC04] Willi Meier, Enes Pasalic, and Claude Carlet. Algebraic attacks and decomposition of Boolean functions. In Christian Cachin and Jan Camenisch, editors, *EUROCRYPT 2004*, volume 3027 of *LNCS*, pages 474–491. Springer, 2004. doi:10.1007/978-3-540-24676-3_28.

[NLV11] Michael Naehrig, Kristin Lauter, and Vinod Vaikuntanathan. Can homomorphic encryption be practical? In *Proceedings of the 3rd ACM Workshop on Cloud Computing Security Workshop*, CCSW '11, page 113–124, New York, NY, USA, 2011. Association for Computing Machinery. doi:10.1145/2046660.2046682.

[SDGM00] Leonie Ruth Simpson, Ed Dawson, Jovan Dj. Golic, and William Millan. LILI keystream generator. In Douglas R. Stinson and Stafford E. Tavares, editors, *Selected Areas in Cryptography, SAC 2000*, volume 2012 of *LNCS*, pages 248–261. Springer, 2000. doi:10.1007/3-540-44983-3_18.

[Üna23] Akin Ünal. Worst-Case Subexponential Attacks on PRGs of Constant Degree or Constant Locality. In Carmit Hazay and Martijn Stam, editors, *EUROCRYPT 2023, Part I*, volume 14004 of *LNCS*, pages 25–54. Springer, 2023. doi:10.1007/978-3-031-30545-0_2.

[YGJL22] Jing Yang, Qian Guo, Thomas Johansson, and Michael Lentmaier. Revisiting the concrete security of Goldreich's pseudorandom generator. *IEEE Transactions on Information Theory*, 68(2):1329–1354, 2022. doi:10.1109/TIT.2021.3128315.