THÈSE DE DOCTORAT DE SORBONNE UNIVERSITÉ

Spécialité

Informatique

École doctorale Informatique, Télécommunications et Électronique (Paris)

Présentée par

Jules Baudrin

Pour obtenir le grade de

DOCTEUR de SORBONNE UNIVERSITÉ

Algebraic properties of symmetric ciphers and of their non-linear components

Thèse préparée au sein de l'équipe-projet Cosmiq d'Inria sous la direction de :

Anne Canteaut Directrice de recherche, Inria, France	Directrice de thèse
Léo Perrin Chargé de recherche, Inria, France	Directeur de thèse

Thèse soutenue à Paris le 6 décembre 2024 devant le jury composé de :

Henri Gilbert Ingénieur général des mines, ANSSI, France	Rapporteur
Sondre Rønjom Professeur, University of Bergen, Norvège	Rapporteur
Christina Boura Professeure, Université Paris Cité, France	Examinatrice
Gohar Kyureghyan Professeure, University of Rostock, Allemagne	Examinatrice
Gregor Leander Professeur, Ruhr University Bochum, Allemagne	Examinateur
François-Xavier Standaert Professeur, UCLouvain, Belgique	Président du jury

Remerciements

Pendant mes études, l'ingénierie informatique, l'architecture, les mathématiques fondamentales *ou* l'enseignement auront tour à tour semblé incontestablement propices à mon épanouissement. Il m'aura cependant fallu onze années pour comprendre que ce n'était pas complètement faux mais que c'était un peu plus compliqué que ça. Tout réside en effet dans l'ambigüité de ce *ou* qui n'a, après tout, aucune raison d'être une disjonction *exclusive* : ces années de thèse en sont le parfait exemple.[1]

Beaucoup ont participé à équilibrer ce juste mélange ; ces quelques remerciements leur sont dédiés.

Je souhaite premièrement remercier Anne et Léo pour leur confiance et leur soutien sans faille depuis le début de mon stage. Merci de votre disponibilité, de nos échanges si enrichissants, et de la liberté de recherche que vous m'avez accordée. Vos approches, très en accord et tout à la fois complémentaires, votre rigueur et votre sens de l'intuition ont et continueront de guider ma façon de chercher. Merci également pour tous les moments hors recherche tout aussi plaisants. Promis, je m'essaie à Emacs d'ici peu.

Merci aux membres du jury, Christina Boura, Henri Gilbert, Gohar Kyureghyan, Gregor Leander, Sondre Rønjom et François-Xavier Standaert d'avoir accepté d'expertiser et de discuter ce travail de thèse. En particulier merci aux rapporteurs, Henri et Sondre, pour votre relecture attentive de ce long manuscrit, qui plus est, en si peu de temps. Merci également aux membres de mon comité de suivi de thèse, Christina Boura et Charles Bouillaguet.

Merci à Christina et à Yann de vous être portés garants lors de ma recherche de stage : cette aventure doctorale aurait été bien différente sans cette impulsion. Merci également pour tous nos échanges pédagogiques, scientifiques, sur l'après-thèse, mais aussi sur le tout et le rien du quotidien.

Merci évidemment à toute l'équipe Cosmiq d'hier et d'aujourd'hui pour votre bonhomie et pour votre support tout au long de ces années. Merci Agathe, André C, André S, Anne, Anthony, Antoine B, Antoine M, Antonio, Augustin, Aurélie, Aurélien, Axel, Bastien, Bruno, Charles, Clara, Clémence, Clément, Cyprien, Daniel, Dounia, Fanny, Ferdinand, Florent, Freja, Gaëtan, Guilhem, Jean-Pierre, Johanna, Justine, Kevin, Laura, Léo, Loïc, Lucien, Magali, Marìa, Matthieu,

[1] À ceux qui se demanderaient où trouver de l'architecture dans ces trois dernières années : précisément dans mon choix d'utiliser un *ou inclusif* et non un *et*. À ceux pour qui ce n'est toujours pas clair, je vous montrerai de jolies photos d'Athènes, de Kobe ou de Pérouse !

Maxime, Merlin, Nicholas, Nicolas D, Nicolas S, Pascale, Paul, Pierre, Quentin, Rémi, Ritam, Rocco, Sacha, Samuel, Simon, Simona, Thomas D, Thomas V, Valentin, Valérian, Virgile. Merci également à `^[A-Z][a-z]+(␣[A-Z])?$` que j'ai évidemment oublié et à tous les membres plus anciens rencontrés en conférence ou à toute autre occasion.

Merci aux buveurs de café du matin, aux cruciverbistes de la pause méridienne, aux affamés du goûter et aux assoiffés des heures non-ouvrées. Un merci tout particulier au groupe des moyens désormais presque tous grands : Augustin, Aurélie, Charles, Clara, Clémence et Nicolas pour votre amitié et pour votre soutien tout au long de cette galère commune. Merci à Augustin pour nos vagabondages à l'étranger, pour notre collaboration sur les machines à pince et pour les sorties course du week-end. Merci à Clara pour nos sorties culturelles, ton initiation au tir à l'arc et pour ton franc parler. Merci à Clémence, éternelle coach de running, pour ton investissement dans tous nos événements collectifs et pour toujours être partante pour des plans aussi fumeux qu'un bowling, un soir de semaine, derrière une station service, au milieu des Pays-Bas. Merci à toutes les deux pour votre amour des potins. Merci à Charles d'avoir refait le monde quotidiennement dans le bureau du fun, pour ta folie communicative, et pour ton soutien jusqu'au dernier clic d'envoi de mon manuscrit. Merci à Nicolas pour ton goût des mèmes de niche et des bonnes bières.

Merci aux deux générations suivantes, Agathe, Aurélien, Axel, Dounia et Virgile d'avoir égayé les journées au labo en parlant tennis, Nintendo, Bourgogne, difficile quotidien d'un conjoint de prof ou d'ordinateur imaginaire... Merci aux plus jeunes encore pour votre nouveau souffle et votre enthousiasme. Merci à mes co-bureaux de la rue Simone Iff, Antoine, Aurélie, Charles, Quentin, Valentin, Valérian, et à ceux de la rue Barrault, Cyprien, Merlin, Thomas et Virgile.

Un grand merci également à Christelle et Christelle pour votre aide dans toutes les démarches.

Merci à l'équipe de Versailles, en particulier, à Christina, Louis, Margot, Pierre, Rachelle et Yann, pour votre accueil toujours chaleureux. Je suis très fier d'avoir pu concourir sous votre drapeau lors de deux classicos de pétanque Inria/Versailles. Merci à Margot, camarade de Master et de thèse pour ta tranquillité et ton art de la présentation. Merci à Pierre pour ton soutien lors des JC2 d'Hendaye et pour nos discussions sur la vie depuis. Merci à Rachelle pour ton amour partagé du Karaoké et de la bière pas chère ! Puissent ses passions s'exporter jusqu'en Belgique.

Merci au groupe Crypto de l'université catholique de Louvain, et tout particulièrement à François-Xavier, pour leur accueil si convivial lors de mes passages ces derniers mois. J'ai très hâte de faire plus ample connaissance avec chacun de vous.

Merci à mes co-auteurs, Anne, Augustin, Christina, Christof, Clara, Gaëtan, Gregor, Lukas, Léo, Matthieu, Nicolas, Pascal, Patrick F, Patrick N, Samuel, Sonia, Thomas et Yann. Écrire ces premiers travaux avec vous fut très enrichissant mais surtout si agréable : cette thèse vous doit beaucoup.

Enfin, merci à tous ceux présents depuis plus longtemps encore. Nolwenn, merci de ton amitié indéfectible depuis 18 ans et d'avoir toujours fait semblant de comprendre mon intérêt pour les maths. Merci à Yumi, pour tous nos moments de vie irremplaçables et de n'être qu'à un SMS d'un match de hand, d'un resto ou d'un apéro trop tardif pour un lundi soir. Merci à JM, éternel fin gourmet, de ton accueil toujours si chaleureux et ton rire contagieux.

Merci à tous les gais-lurons d'Epita qui n'ont eu d'autre choix que de supporter mon humour vaseux pendant deux ans et demi, tout en bravant le froid du scooter, les intarissables réapprovisionnements sucrés du midi, les salles-machines surpeuplées et les Villejuif nights plus que douteuses. En particulier merci à Alexandre, Antoine, Arthur, Cyril, Hadrien, Jérémy, Manuel, Maxence, Nicolas C, Nicolas R, Romain pour tous ces bons moments et pour tous nos verres depuis. Merci au *FC Derank* de calmer mes ardeurs vidéoludiques.

Merci à Vasco pour notre complicité sur les bancs de la fac, du 2bis et de toutes les pizzerias de Paris. Merci à Victor, pour tes feus fromages, mais surtout pour ton appétence pour le cryptogramme de La Buse et autre manuscrit de Voynich qui ferait de toi un zélé cryptographe. Abel, malgré tes choix très discutables en matière de factions dans les RTS et de passions mathématiques, je suis très heureux de te retrouver à Bruxelles d'ici peu.

Merci à ceux présents depuis toujours : mes parents, pour leur soutien indéfectible, Nathan et Gaspard pour nos tennis salvateurs, Marie, Emil et Florian pour les pensées protectrices envoyées d'outre-Rhin.

Enfin, merci à Pauline qui, parce qu'elle a eu le malheur de faire des maths, a quotidiennement dû avoir affaire à des "mais si tu sais, les corps finis, les polynômes, la transformée de Fourier" et qui, en toute connaissance de cause, a tout de même tenu à relire ce manuscrit intégralement. Merci pour ton soutien, ta patience, ton amour, pour toutes ces belles années et toutes celles qui viendront.

Contents

Notation

$\mathbb{N}$	The set of natural integers.
$n \in \mathbb{N}$	The number of variables.
$m \in \mathbb{N}$	The number of coordinates (if $n \neq m$).
$\lvert Z \rvert$	The cardinality of a set Z.
$\langle x, y, z \rangle$	The subspace spanned by x, y, z.
$\{\!\{x, x\}\!\}$	The multiset containing x with multiplicity 2.
$\mathbb{F}_2$	“The” finite field with two elements: $(\{0, 1\}, +, \cdot)$.
$\mathbb{F}_2^n$	The canonical $\mathbb{F}_2$-space of dimension n.
$\mathbb{F}_{2^n}$	“The” finite field with 2^n elements.
x^u, X^u	The products $\prod_{i=0}^{n-1} x_i^{u_i}$, $\prod_{i=0}^{n-1} X_i^{u_i}$.
$x^{(0)}, \dots, x^{(t)}$	A sequence of values x changing in “time”.
$x_0, \dots, x_i$	The coordinates of a fixed value x.
$\left(x_i^{(t)}\right)^{\ell}$	The ℓ-th power of the i-th coordinate of $x^{(t)}$.
$[\![i, j]\!]$	The set of integers ranging from i to j, both included.
$\mathbf{1}_Z$	Indicator function of a set Z.
$\mathbf{1}_u$	Indicator function of a singleton: $\mathbf{1}_u := \mathbf{1}_{\{u\}}$.
$\mathrm{Supp}(u)$	The support of u: $\mathrm{Supp}(u) := \{i \in [\![0, n-1]\!],\ u_i = 1\}$.
$u \preceq v$	The covering relation $u \preceq v \iff \mathrm{Supp}(u) \subset \mathrm{Supp}(v)$.
χ_u	Character associated to u: $\chi_u \colon \mathbb{F}_2^n \to \mathbb{C}, x \mapsto (-1)^{u \cdot x}$.
f, g, h	Some Boolean functions.
F, G, H	Some vectorial Boolean functions.
$u \cdot v$	Standard dot product $u \cdot v := \sum_{i=0}^{n-1} u_i v_i$.
W_f, W_F	Walsh transform $W_f \colon \mathbb{F}_2^n \to \mathbb{Z}$.
$\mathcal{L}(F), \delta_F$	The linearity and differential uniformity of F.
$Z_F^{\mathrm{diff}}(\Delta^{\mathrm{in}}, \Delta^{\mathrm{out}})$	The set of solutions of a differential equation.
$Z_F^{\mathrm{lin}}(\alpha, \beta)$	The set of solutions of a linear approximation.
$Z_F^{\mathrm{comm}}(A, B)$	The set of solutions of a commutative equation.
$\sim_{\mathrm{lin}}, \sim_{\mathrm{aff}}, \sim_{\mathrm{EA}}, \sim_{\mathrm{CCZ}}$	The linear, (ext.) affine, and CCZ equivalence relations.
$\sim_{\mathrm{sim}}$	The similarity equivalence relation.
$\mathbf{T}^F, \mathbf{C}^F, \mathbf{D}^F$	The transition, correlation, quasi-differential matrices.
$\mathbf{M}_{n \times m}(\mathbb{F}_{2^k})$	The set of $n \times m$ matrices with coefficients in $\mathbb{F}_{2^k}$.
$\mathbf{GL}_n(\mathbb{F}_{2^k})$	The set of $n \times n$ invertible matrices with coeff. in $\mathbb{F}_{2^k}$.

Chapter 1

Introduction to symmetric cryptography

Contents

1.1 Overview

The term *cryptography* refers to the study and practice of the techniques providing secrecy in the presence of untrusted parties. Such scenarios in particular include the case where a person stores data that should remain confidential, or where two (or more) people would like to privately communicate.

The most ancient forms of cryptography date back to at least 1500 BC [Kah96]. However, with the emergence of computers and the World Wide Web, the need for confidentiality became a daily matter, with a gigantic amount of data stored and exchanged each day.

In order to achieve such secrecy, any cryptographic technique must rely on secret data that are usually called the *key*, known only by (part of) the legitimate users. This key is used to transform the original data, known as *plaintext*, into an unintelligible message called *ciphertext*. This *encryption* operation must be reversible, so that genuine information can be recovered, but only by people having the knowledge of the key. The inverse transformation is known as *decryption*.

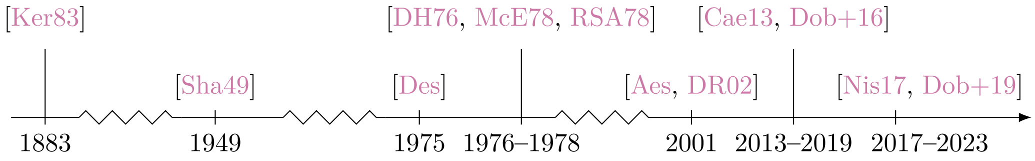


Figure 1.1: A few milestones of modern cryptography.

When the same key is used for both encryption and decryption, these methods are described as *symmetric*, thus emphasizing the common knowledge shared by the sender and the recipient. These techniques however presuppose the ability of the protagonists to agree on such a shared secret. Because Internet exchanges are remote and instantaneous, techniques enabling secure transmissions *without preceding communication* are necessary. These methods rely on a *private key* chosen by a user A, but also on a *public key* derived from it. The public key is made available to anyone and is used to encrypt messages addressed to A. On the other hand, only the knowledge of the private key enables to decrypt the ciphertext. Because of the distinct keys playing distinct roles, such methods are known as *asymmetric* primitives.

While asymmetric cryptography solves more challenges than symmetric cryptography, it is also in practice way slower. In most of the use cases, asymmetric and symmetric techniques are therefore used in a *hybrid* way: first, an asymmetric scheme is used to handle the first exchanges, and in particular the *key-agreement*; only then is a symmetric scheme used to encrypt the following exchanges in an effective manner.

1.2 A few milestones of modern cryptography

Kerckhoff's principle. While the first mathematical treatment of cryptography by Shannon [Sha49] is arguably the turning point from "historical" to "modern" cryptography, the most essential principle followed by all *academic* cryptographers was stated 60 years before by Kerckhoffs [Ker83], among six requirements for military cryptography:

> «Il faut que [le système] n'exige pas le secret, et qu'il puisse sans inconvénient tomber entre les mains de l'ennemi.»

Kerckhoffs thus warns that the security of any cryptographic system should rely on the secret key *only*. In particular, the secret of the scheme itself should not be considered while analyzing cryptographic resistance.[1] Today, it is a common practice to assume that any opponent knows the system in detail while *cryptanalyzing* a cipher.

[1] This is in line with Murphy's law which in this case could be interpreted as "the description of a secret scheme will eventually leak".

Shannon's mathematical theory of cryptography. In line with Shannon's work, such an analysis of a cipher is today done mathematically. From a theoretical point of view, *unconditional security* is the ultimate confidentiality level, as it states that no information about a plaintext can be obtained from the corresponding ciphertext. However, Shannon proved that for unconditional security to hold, the key must necessarily be at least as long as the encrypted message. For practical reasons, such a key length is (almost) never reached. That is why the security of a cryptographic primitive is instead measured by weighing how much computational effort, memory storage and input/output data are needed for an attacker to *break* a cipher, *i.e* to gain access to some information about the key or the message. However, cryptanalysis today is only able to provide insecurity proofs, by mounting attacks, but not actual security proof. In the case of asymmetric schemes, the security is therefore studied through *reductions proofs* which point out that breaking a specific cipher is at least as hard as breaking some well-studied mathematical problems that is *considered* really hard. As highlighted by Chapters 3 to 5, in the case of symmetric schemes, the assessment of the *practical security* is done by a continual analysis of the primitives, or of simplified versions of it. On the other hand, the security of a *mode of operation* which provides guidance on how to properly encrypt long messages, can be proven if we *assume* that its building blocks behave *ideally*.

Birth of asymmetric cryptography. Since Shannon, modern cryptography really took a decisive turn with the invention of the already-mentioned asymmetric cryptography by Diffie & Hellman [DH76]. They presented in 1976 the first key-exchange protocol, which paved the way to the first actual asymmetric cryptosystems suggested by McEliece [McE78] on the one hand, and by Rivest, Shamir and Adleman [RSA78] on the other hand. While the latter cipher, which is known as RSA due to the initial letters of the name of its designers, is without a doubt one of the asymmetric primitives that is nowadays the most used, the one suggested by McEliece is among the most likely to resist the threat caused by quantum computing [Nis22].

Standards over the years. Since the 1970s, the usage of cryptography has then been driven by standardization. Because of the endless process that is cryptanalysis, focusing efforts only on some cryptosystems enables the community to gain *more* trust in only a *few* primitives. It also has a practical reason for being: people need to use the same algorithms in order to communicate. Depending on specific needs and/or constraints, the usage is therefore guided by national or international legal regulation. These local *de jure* standards, often lead to global *de facto* ones when the usage of a specific cipher becomes dominant. In symmetric cryptography, the first major standard is the American Data Encryption Standard [Des], known as DES. DES was designed by IBM in collaboration with the NSA after a call for proposals from the American National Bureau of Standards in 1972 and standardized in 1977.

In 2000, The National Institute of Standards and Technology of the United States of America (NIST) promoted the successor of the DES: the Advanced Encryption Standard [Aes], which is known as AES. Originally named Rijndael [DR02] by its academic creators, AES was adopted after a 3-year process of design and analysis which also involved international academics and companies. It is today the most widely-used symmetric primitive.

More recently, many standardization processes were launched by other actors, such as the NESSIE [Nes00] project of the European Commission, the eSTREAM [Ecr04] project of the ECRYPT network, the recommendations of the Japanese CRYPTREC committee, or the CAESAR [Cae13] international academic competition. Regarding symmetric primitives, the most recent initiative is the NIST process for *lightweight* symmetric cryptography [Nis17], whose goal is to ensure security of very-constrained devices such as *connected objects*. It ended in 2023 and selected Ascon [Dob+21] as its future standard [Dob+19]. Ascon was already chosen in 2019 within the “lightweight applications (resource constrained environments)” portfolio of the CAESAR competition [Dob+16].

1.3 Symmetric cryptography

This thesis deals with the security analysis of some of the main components used in symmetric cryptography. Before presenting the main contributions in Section 1.5, this section is dedicated to an overall description of the targeted security notions, of the main objects at hand to achieve these notions, and finally of the main analysis methods to assess the resistance of a scheme.

1.3.1 Security notions

1.3.1.a Confidentiality

When it comes to cryptography, the main challenge is to ensure the *confidentiality* of a private communication. In other words, even if a ciphertext is intercepted by an eavesdropper, it should not leak any information on the corresponding ciphertext, either without any other assumption in the case of unconditional security, or with limited resources in the case of computational security.

In order to get a better grasp of this definition, let us introduce the formalism and some notation that will be used throughout this manuscript. First, because modern cryptography is concerned with computer usage, it is always assumed that both the plaintext x, the ciphertext y and the secret key k are finite sequences of bits. It should *a priori* be possible to encrypt any sequence of *any finite length*. For now, let us consider that the size of the plaintexts and ciphertexts are *fixed*. We denote by $n \in \mathbb{N} \setminus \{0\}$ the *block size*, that is, the length of the plaintext and the ciphertext, and by $\kappa \in \mathbb{N} \setminus \{0\}$ the *key size*, so that x, y, k satisfy:

$$x \in \{0,1\}^n , \quad y \in \{0,1\}^n , \quad k \in \{0,1\}^\kappa .$$

From there, the central object at hand can be defined as follows.

Definition 1.1 (Block cipher)**.** Let $n \geq 1$, $k \geq 1$. A *block cipher* $\mathcal{E}$ is a sequence $\mathcal{E} = (E_k \colon \{0,1\}^n \xrightarrow{\sim} \{0,1\}^n)_{k \in \{0,1\}^\kappa}$ where for any $k \in \{0,1\}^\kappa$ the function E_k is bijective. Stated otherwise, a block cipher is a family of 2^κ bijections over $\{0,1\}^n$. For a fixed $k \in \{0,1\}^\kappa$, the bijection E_k is referred to as an *instantiated cipher.* ▷

The concept of block cipher is the most natural way to make the definitions of Section 1.1 concrete. Indeed, encryption associates a plaintext to a ciphertext in a revertible manner: this is the role of a bijective function. Furthermore, encryption should depend on the secret, so multiple bijections indexed by the actual chosen key should be considered. However, we can already note that a block cipher alone does not address all the problems raised by cryptography, as it only enables the encryption of *fixed-sized messages.* Furthermore, the definition is free from any security consideration, but we can now properly characterize perfect secrecy.

Definition 1.2 (Perfect secrecy)**.** Let $\mathcal{E} = (E_k \colon \{0,1\}^n \to \{0,1\}^n)_{k \in \{0,1\}^\kappa}$ be a block cipher. Let K be a uniform random variable over $\{0,1\}^\kappa$. The block cipher $\mathcal{E}$ is *perfectly secure* if for any random variable X over $\{0,1\}^n$, the random variable C defined by $C = E_K(X)$ is independent of X. Equivalently, $\mathcal{E}$ is perfectly secure if for any $x_0, x_1, y \in \{0,1\}^n$ it holds that:

$$\mathbb{P}\left[E_K(x_0) = c\right] = \mathbb{P}\left[E_K(x_1) = c\right].$$

▷

The following example is due to Shannon [Sha49].

Example 1.3 (One Time Pad)**.** Let $n \geq 1$. Let us consider the block cipher $\mathcal{E}$ with an n-bit block size and and an n-bit key size that is defined as follows:

$$\forall\ k \in \{0,1\}^n, \forall\ x \in \{0,1\}^n, \quad E_k(x) = x + k,$$

where the addition is a coordinate-wise modulo 2 addition. In other words, if $x = (x_0, \ldots, x_{n-1})$ and if k and $y := E_k(x)$ are decomposed similarly, it holds that:

$$\forall\ i \in [\![0, n-1]\!], \quad y_i := x_i + k_i.$$

Let us consider that k is picked uniformly at random. For a fixed bit of plaintext x_i, we observe that x_i is encrypted as 0 if and only if $k_i = x_i$ and as 1 if and only if $k_i = x_i + 1$. In particular, both events hold with probability $\frac{1}{2}$, and this is independent of the actual value of x_i. This is the reason why the block cipher $\mathcal{E}$, that is known as the *One Time Pad* (OTP) or as the *Vernam cipher*, is perfectly secure. ▷

While theoretically secure, we observe that the One Time Pad is not optimal when it comes to implementation. Indeed, its key is as long as the plaintext. When the key agreement is done online, this presupposes that an n-bit long message (the key) has already been securely exchanged, so it could as well have been the plaintext. The whole symmetric encryption becomes in that case meaningless.

For such reasons, we would like the key to be strictly shorter than the encrypted messages, or equivalently, we would like the key to be reusable for future encryptions. This is however not possible for OTP. Indeed, let us assume that an adversary knows for a fact that an observed ciphertext y corresponds to the encryption of a known plaintext x. Such a situation is known as a *known-plaintext scenario*. In that case, the attacker can compute $x + y$ and instantly recovers the key k. This implies that for any future eavesdropped ciphertext y', the corresponding plaintext x' can be recovered by computing $x' = y' + k$. If the key is reused, OTP then suffers from a strong confidentiality issue.

It is then necessary to find ways of designing block ciphers that can be effectively implemented and for which security can be assessed. The main design methods are presented in Section 1.3.2, while cryptanalysis is addressed in Section 1.4.

1.3.1.b Integrity and authenticity

The sole notion of confidentiality is not the only desired security notion. For instance, one can also expect the *authenticity* of a received message. This corresponds to the guarantee of receiving messages from the *expected* sender and not a malevolent one. Authenticity cannot be achieved by the use of a block cipher alone. Indeed, let us consider a *known ciphertext setting*, that is, a scenario where a *passive* adversary only observes exchanged ciphertexts. In that case, as long as the key used to encrypt the messages does not change, the attacker can break authenticity by sending again any of the observed ciphertexts because the recipient has no way of verifying the authenticity of the message. Even when used in a mode of operation, this is not the primary role of an encryption scheme. One way to solve this is to consider a message authentication code (or MAC).

Definition 1.4 (Message authentication code (MAC))**.** Let $\kappa \geq 1, n \geq 1$. A message authentication code is a function $\mathcal{F}\colon \{0,1\}^\kappa \times \{0,1\}^\star \to \{0,1\}^n$, where the star $\star$ highlights the fact that the second input is a bit sequence of *any* finite length. Equivalently, the MAC $\mathcal{F}$ can be seen as a family of functions $F_k\colon \{0,1\}^\star \to \{0,1\}^n$, where $F_k = \mathcal{F}(k, \cdot)$ for any k. ▷

A MAC is designed to produce a finite tag T of length n from the secret key $k \in \{0,1\}^\kappa$ and from either the plaintext or the ciphertext $x, y \in \{0,1\}^\star$. The definition is again abstracted from security consideration, but it should be hard in practice to output a valid tag T for any chosen plaintext x without knowledge of the actual used key. This is called a *forgery*.

Message authentication codes are also a way to ensure *integrity* of a message. In other words, it enables the recipient to verify whether or not the original message has been tampered. We can for instance think about OTP where an *active* attacker can flip any bit of a ciphertext y during communication. This ultimately leads to a tampered message x where the corresponding bit is flipped. Such properties reflect the so-called *malleability* of the cipher.

The role of encryption schemes and MAC are in practice less compartmentalized than this introduction might suggest. For instance, *authenticated encryption* (AE) schemes (with associated data, known in that case as AEAD schemes) are today designed to provide integrity, authenticity and confidentiality in an all-in-one manner [BN00, Rog02]. This is for instance the case of Ascon that is described and analyzed in Chapter 3. The close relations between the two kinds of primitive are also highlighted in Chapter 7, and more precisely in Section 7.1, where two MACs are designed based on the round function of the AES block cipher.

1.3.2 Standard ways of designing a symmetric cipher

1.3.2.a From theory to practice

Basic attacks and sizing. As presented above, a block cipher is on paper a simple family of bijections. However in practice, both the block size n and the key size κ should be large. Indeed, if n is too small, then an attacker can query the online encryption of the 2^n possible plaintexts, or wait and intercept a lot of known plaintext/ciphertext pair, until he/she builds and *stores* the *full code-book*. This way, as long as the key is not changed, the confidentiality is compromised. Such an attack can never be excluded. For this reason, the block sizes of most of the recent block ciphers are usually[2] strictly higher than 64 bits, as 2^{64} is considered a very high order of magnitude in the case of memory storage. Instead, when the key size κ is too small, then from a known plaintext/ciphertext pair (x, y), an adversary can encrypt x with all the 2^κ possible keys k', and reduces the key space to only the ones such that $E_{k'}(x) = y$. This is known as the *brute force attack*. This time, this is the computational power of the attacker that guides the usual sizing and we expect that a key space of cardinality 2^{128}, *i.e.* $\kappa = 128$ is sufficient to exclude such an attack. In the recent years, a key size of $\kappa = 256$ is considered sufficient to resist the threat of quantum computing. These are *gigantic* numbers which are impossible for a Human being to perceive. For example, 2^{64} is about a thousandth of the estimated radius of the Milky Way in centimeters (around 2^{75}). The value 2^{256} is about a thousandth of the estimated number of atoms in the universe.

Computer-oriented constraints. With these points of reference in mind, it is clear that a random family of bijections cannot be chosen to be a practical block cipher. Indeed, the table of values of a *single* 64-bit bijection $F\colon\ \{0,1\}^{64} \to \{0,1\}^{64}$, or even an optimized implementation of it, is already hardly impossible to store on a computer. Furthermore, we have not yet taken into account that encryption (and possibly decryption) should be very effective. Likewise, security should also be possible to assess and cryptographers should be able to analyze the properties of an enormous number of functions at once. Finally, because money is the sinews of war, encryption should of course be done at the lowest cost possible, where the measurement of cost is often power or energy consumption. These three criteria, security, performance, and cost, are the main ones used to compare different

[2] As the growing interest for memory encryption or cache randomization highlights, this is not always the case. The tweakable block cipher SCARF [Can+23] has a 10-bit block size but is designed for a very specific use case.

cryptographic constructions, which always come with a unique trade-off between the three.

Lightweight cryptography. These trade-offs have recently opened many questions, especially when it comes to extremely-constrained devices. Indeed, the number of devices which work for instance on battery, and whose primary purpose is not security is exploding. However, in many scenarios, cryptography is still necessary to ensure the proper functioning of such objects. This is the case for example of healthcare devices such as pacemakers, of "smart home" gadgets, but also of the now widespread RFID tags. These new use cases then bring their share of constraints, which can take many different forms [BP17], and which are hard to categorize under a sole definition. Yet, *lightweight primitive* is the umbrella word that covers this new trend in symmetric cryptography. The block ciphers Ascon and Midori that are respectively presented and studied in Chapter 3 and Chapters 4 and 5 belong to this class of designs.

1.3.2.b Block ciphers

Iterated constructions. Because of the computer-oriented constraints presented above, designing a block cipher is equivalent to *effectively providing* 2^κ random-looking bijections. To do so, the problem is often bypassed by using an *iterated construction.*

Construction 1.5 (Iterated construction). *Let $\kappa \geq 1, n \geq 1, R \geq 1$. An R-round iterated block cipher $\mathcal{E} = (E_k \colon \{0,1\}^n \xrightarrow{\sim} \{0,1\}^n)_{k \in \{0,1\}^\kappa}$ is defined using:*

- *a* key schedule *which is a function* $\mathrm{KS} \colon \{0,1\}^\kappa \to (\{0,1\}^n)^{R+1}$*, and*
- *a* round function *which is a bijection* $F \colon \{0,1\}^n \to \{0,1\}^n$.

Let $k \in \{0,1\}^\kappa$. From k, $R+1$ rounds keys *are derived using the key schedule:* $\mathrm{KS}(k) := (k^{(0)}, \dots, k^{(R)})$*. Then, the instantiated cipher E_k is defined by:*

$$E_k = T_{k^{(R)}} \circ F \circ T_{k^{(R-1)}} \circ F \circ T_{k^{(R-2)}} \circ \ldots \circ F \circ T_{k^{(0)}},$$

where for any $c \in \{0,1\}^n$, the function T_c is the addition of c modulo 2 coordinate by coordinate: $T_c \colon x \mapsto x + c$. This construction is also called a key-alternating cipher*.*

This generic construction is depicted in Figure 1.2. While being a very particular case of Definition 1.1, an iterated construction has huge advantages. First, it reduces the problem of defining 2^κ bijections into the definition of only two functions: KS and F. It is also intended to reach a random-looking behavior after a sufficient number of iterations of the round function followed by a round-key addition. This is of course not true in general and should be carefully analyzed. As we will see throughout this manuscript, this assumption is valid under some hypotheses, but also fails spectacularly in some cases which inevitably lead to security flaws.

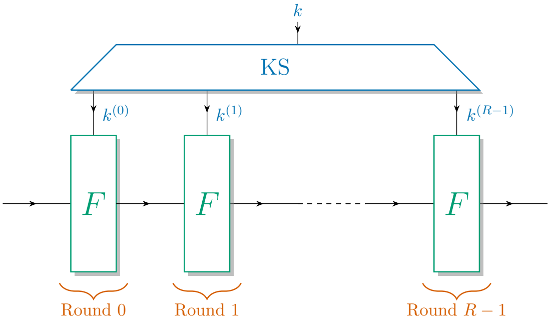


Figure 1.2: A key-alternating cipher.

Nonetheless, as shown in Section 1.4, the other strong advantage of an iterated construction is that it can be thoroughly cryptanalyzed.

Remark 1.6 (Cryptographic functions). Construction 1.5 is an iterated construction of block ciphers. Many other cryptographic objects can be built using a similar approach. This is the case of *cryptographic hash functions* which are primitives mapping an arbitrary-long sequence of bits to a fixed-size one. Among their many usages, the most iconic one is password hashing: instead of storing the plain password of a user on a server, it is a good practice to instead store the image of the password by a *one-way function*, which can be a hash function, or a *password hashing algorithm*. This way, if the server is compromised, the original password is not. This of course holds if the hash function is "hard to inverse". This hardness is measured using the notion of (second) pre-image resistance and collision resistance. Hash functions are not studied in this thesis, but the notion of *universal hash function* introduced and studied in Section 7.1 is related, yet the involved security notions are different.

Cryptographic permutations are also very useful. They are random-looking *public* n-bit bijections which can be iteratively built. In that case, they can be thought of as an instantiated cipher E_k where k is publicly known. They are useful for instance to build hash functions, or AEAD encryption schemes, as it is highlighted in Section 3.2. ▷

Thanks to Construction 1.5, we can now focus on the design of a bijective round function. We present in the following the two main constructions that are used in practice.

Feistel networks. The first construction is the *Feistel network*. The first famous usage of this structure appears in the block cipher Lucifer designed by IBM as a precursor of the DES, which is also based on such a construction. The principle of a Feistel network is to reduce the problem of building a $2n$-bit bijection, to only building an n-bit *function*, by using the structure described in Figure 1.3. This structure has strong advantages. Indeed, as we will see in Chapter 6, it is often easier to build functions with optimal properties (with respect to some specific kind of attacks) than to build optimal bijections. Furthermore, it is also strongly favorable regarding both software and hardware implementations: the only look-up table or printed circuit board to consider is the one corresponding to the function S with n input and output bits, which is significantly smaller than the one of the round function F over $2n$ bits. Indeed, the wire swap is just a matter of appropriate physical wirings or of a logical swap of variables. Such constructions are still often used today. This is for example the case of the cipher Simon [Bea+13] designed by the NSA or its academic twin Simeck [Yan+15].

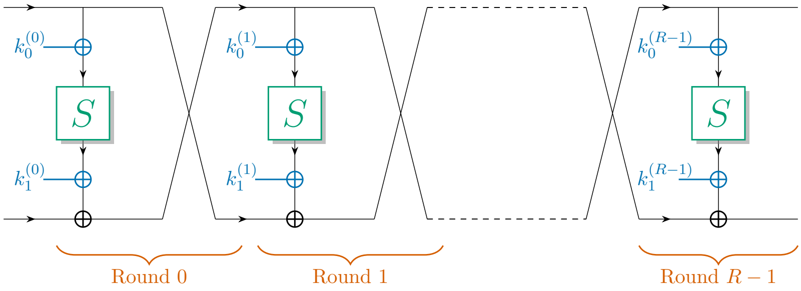


Figure 1.3: A Feistel network.

Substitution permutation networks. The approach taken by the so-called *substitution permuation networks* (SPN) is slightly different. It reduces the problem of building an n-bit bijection into the problem of building an m-bit bijection S, where m is a (small) divisor of n, and an n-bit linear bijection L. As highlighted in Figure 1.4, the so-called *substitution box* S, and usually referred to as *Sbox*, is used $\frac{n}{m}$ times in parallel, while the linear bijection can *a priori* shuffle the full n-bit state. This is again driven by implementation reasons. Indeed, as explained in more detail in Chapter 2, a block cipher should be non-linear. However, non-linear functions are in practice costly to implement. With such a construction, the round function F is non-linear, but the only non-linear component that is involved is of small size, and therefore has (in comparison) a smaller circuit area or a smaller table.

Note that both Feistel networks and SPN follow a rule of thumb identified by Shannon [Sha49]. In its own words, a secure cipher should “frustrate” a statistical analysis by using both a *diffusion* method and a *confusion* one. Diffusion corresponds to the idea that many input bits and key bits should be involved in the expression of each output bit, while confusion reflects the fact that these

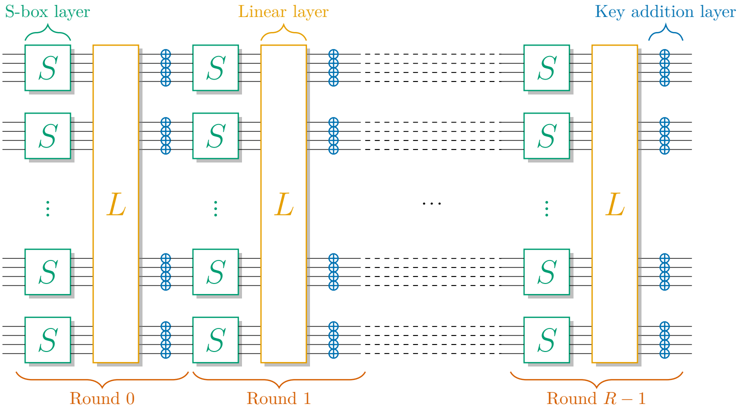


Figure 1.4: A Substitution permutation network (SPN).

dependencies should be as intricate as possible. In the previous constructions, diffusion is operated by global linear layers, while confusion is obtained through local non-linear operations.

The most renown example of SPN is of course the *de facto* standard, *i.e.* AES. Its round function is briefly described in the following example.

Example 1.7 (Round function of the AES.)**.** The AES is a 128-bit state block cipher. Its state is usually represented as 4×4 matrix of bytes, whose entries are numbered from top to bottom, and from left to right. This is highlighted in Figure 1.5. The round function $F\colon \{0,1\}^{128} \to \{0,1\}^{128}$ of AES follows an SPN construction. It can indeed be decomposed as follows:

0	4	8	12
1	5	9	13
2	6	10	14
3	7	11	15

Figure 1.5: The matrix representation and bytes numbering of the state of AES.

$$F := \mathsf{MC} \circ \mathsf{SR} \circ \mathcal{S},$$

where $\mathsf{MC}, \mathsf{SR}, \mathcal{S}$ are respectively called the MixColumns, the ShiftRows and the Sbox layers. These different layers are depicted in Figures 1.6 and 1.7.

Let us describe each of them more precisely.

Sbox layer. The Sbox layer consists in the parallel application of a single 8-bit bijective Sbox S on each of the byte of the state. The Sbox is explicited later in Section 2.3.2.c.

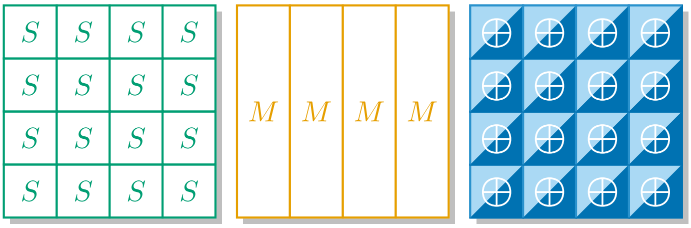


Figure 1.6: The Sbox, MixColumns and constant addition layers of AES.

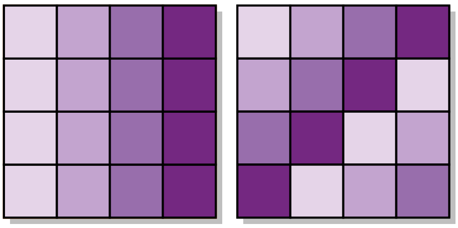


Figure 1.7: The state of AES before (left) and after (right) the ShiftRows layer.

ShiftRows layer. The ShiftRows operation is a reorganization of the bytes of the state: for any $i \in [\![0, 3]\!]$, the i-th row (starting from the top) is cyclically shifted by i positions to the left.

MixColumns layer. The MixColumns layer is the parallel application of a single 32-bit linear bijection M on each column of the state. More specifically, M is $\mathbb{F}_{2^8}$-linear and can then[3] be represented as a matrix-vector multiplication. *When the field with 256 elements is represented as* $\mathbb{F}_{2^8} := \mathbb{F}_2[X]/(X^8 + X^4 + X^3 + X + 1)$, M has the following matrix representation:

$$M := \begin{pmatrix} \texttt{0x2} & \texttt{0x3} & \texttt{0x1} & \texttt{0x1} \\ \texttt{0x1} & \texttt{0x2} & \texttt{0x3} & \texttt{0x1} \\ \texttt{0x1} & \texttt{0x1} & \texttt{0x2} & \texttt{0x3} \\ \texttt{0x3} & \texttt{0x1} & \texttt{0x1} & \texttt{0x2} \end{pmatrix}.$$

Round key addition. The round function F is followed by the addition of a 128-bit round key derived from a key-schedule algorithm. We refer to the recent work of Leurent & Pernot [LP20] regarding its original and alternative descriptions.

In order to match with the description made in Figure 1.4, the linear layer L is then defined by $L :=$ MC $\circ$ SR.

▷

[3] The necessary notions about finite fields are covered in Chapter 2.

Modes of operation. Finally, since the beginning of Section 1.3.1, we restricted ourselves to the encryption of messages of fixed size n. This is of course a strong constraint. It seems natural in that case to cut a long message into n-bit blocks and use a block cipher multiple times. This first points out the question of *padding* in order to handle the encryption of the tail of the message which can be of length strictly less than n. But the more important problem is that the naive successive usages of a block cipher with a fixed key cannot ensure confidentiality. This is for instance highlighted by the now viral example of the "ECB Penguin" [EU06]: if each pixel of an image is independently encrypted, the encrypted image is only a copy of the original one, only with different colors. However, there exist ways [RBB03, MV04, Dwo07, Dwo04] of properly using a block cipher with a fixed key to encrypt long messages. Those methods, including the insecure ECB, are called *modes of operation* and their *provable security* is well-studied by assuming that the block-cipher being used behaves randomly.

1.3.2.c Stream ciphers

From OTP to stream ciphers. There also exists another large class of ciphers which are not (necessarily) based on a block cipher. These ciphers are known as *stream ciphers* and can be considered as a natural adaptation of Example 1.3 that makes it efficient in practice. Indeed, OTP has by design a strong advantage: its construction works for *any arbitrary plaintext length* and therefore does not need any mode of operation. Its main problem that needs to be solved is that the random key k, which is used as a *mask* for the plaintext x, needs to be replaced after a single use. Instead, this random mask could be replaced by a pseudorandom one. Stated otherwise, one way of adapting OTP into a practical cipher is to replace the mask generation by a function which takes as input the key k and which outputs an arbitrary-long pseudorandom mask z. This leads to the following definition of a stream cipher.

Definition 1.8 (Stream cipher)**.** A stream cipher is a function $F\colon \{0,1\}^\kappa \times \{0,1\}^s \to \{0,1\}^\star$ which takes as input a κ-bit key and an s-bit initial value and produces an (arbitrary) long sequence $z = \left(z^{(t)}\right)_{t\in\mathbb{N}} \in \{0,1\}^\star$. ▷

The initial value, which is usually called IV, is a public value whose purpose is to be changed very frequently, so that the key can be used for a longer time. With a stream cipher, the encryption and decryption of the t-th bit $x^{(t)}$ of plaintext works exactly in the same way as with OTP. More precisely, the corresponding ciphertext bit $y^{(t)}$ is computed as $y^{(t)} = x^{(t)} + z^{(t)}$ and decrypted using the fact that $x^{(t)} = y^{(t)} + z^{(t)}$. Contrary to the case of a block cipher, a stream cipher inherently handles the encryption of arbitrarily long sequences of plaintext. However, as for block ciphers, the definition does not take into account any security argument, and no guidance is given on the actual way of deriving the arbitrarily long sequences from the key and IV.

Security arguments. The security of such constructions almost always sums up to weighin the randomness of the sequence z. Indeed, in a known-plaintext scenario, the knowledge of $x^{(t)}$ and $y^{(t)}$ is sufficient to recover $z^{(t)}$ by computing $z^{(t)} = x^{(t)} + y^{(t)}$. From there, any statistical argument can be used to distinguish the sequence z from a random sequence. Among the most common methods, correlation attacks [Sie84, Sie85] try to detect correlation between multiple bits of a sequence z generated from a fixed pair (k, IV), or correlations between bits of multiples sequences $z^{(0)}, \dots, z^{(\ell-1)}$ output from a fixed key k for different initial values. When instead, correlation can be detected between output bits and the key, key recovery attacks can be mounted. One way of mounting such attacks is to study the algebraic representation of each output bit. This is the approach taken by the so-called *cube attacks* [Vie07, DS09] that are studied to a great extent in Chapter 3, but in the context of block ciphers. Fast correlation attacks [MS89, CT00] tackle the problem from a coding theory perspective. In that case, the ciphertext is considered as a *noisy* plaintext, where the noise is the mask z, and the goal is then to find a way to decode it into its genuine form x.

Generic constructions. Regarding generic constructions, the most common way of building a stream cipher is to consider an internal s-bit state (or *register*), and three functions of the following form:

$$I\colon \{0,1\}^k \times \{0,1\}^s \to \{0,1\}^r, \quad g\colon \{0,1\}^r \to \{0,1\},$$

$$F\colon \{0,1\}^r \times \{0,1\}^k \times \{0,1\}^s \to \{0,1\}^r.$$

The initial value of the state is computed as $X^{(0)} = I(k, IV)$. Then the state is updated. The value of the register at clock $t+1$ is computed from its current value, the key and the IV. More precisely $X^{(t+1)}$ is defined by $X^{(t+1)} := F_{k,IV}\left(X^{(t)}\right)$. Finally, at each clock, a bit of mask is output and it is computed as $z^{(t)} := g\left(X^{(t)}\right)$. This is depicted in Figure 1.8.

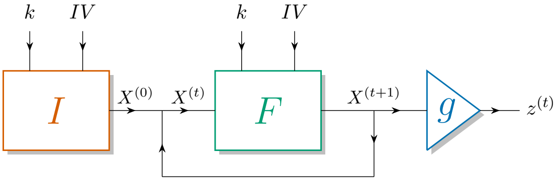


Figure 1.8: A generic stream cipher.

With such a construction, the same routine is used at each step, which greatly simplifies the implementation and (sometimes) the cryptanalysis.

Finally, even if Definition 1.8 is defined over the alphabet $\{0,1\}$, it should be noted that the definition of a stream cipher can be adapted to any group $(\mathbb{G}, +)$. In that case, encryption is still the addition (with respect to a given addition law) between the plaintext $x^{(t)} \in \mathbb{G}$ and the mask $z^{(t)} \in \mathbb{G}$, while decryption becomes the addition with the opposite $-z^{(t)}$ of the mask $z^{(t)}$. While stream ciphers are not the main topic of this thesis, we still discuss them in more detail in Section 7.2 where the design and initial analysis of a stream cipher over a finite field with an odd number of elements are presented.

1.4 Analysis of symmetric ciphers

Let us now develop a bit more the analysis of symmetric ciphers, which is our main concern in Chapters 3 to 5. As already mentioned, apart from implementation considerations, iterated constructions are also more-easily studied than *ad-hoc* constructions. In particular, the properties of the round function can be thoroughly studied. As a matter of fact, from the point of view of Kolmogorov [Kol98], a 128-bit bijection which can be implemented in practice is strictly the opposite of a random function. This is where theory and practice collide. This is also the reason why *indistiguishability* is a first good measurement of the security of a block cipher.

Definition 1.9 ((Informal) Indistinguishability)**.** Let $\mathcal{E} = (E_k \colon \{0,1\}^n \xrightarrow{\sim} \{0,1\}^n)_{\{0,1\}^\kappa}$ be a block cipher. The block cipher $\mathcal{E}$ is considered *indistinguishable* if the uniform draw $E_k \overset{\$}{\leftarrow} \mathcal{E}$ cannot be distinguished from the uniform draw $F \overset{\$}{\leftarrow} \mathrm{Bij}(\{0,1\}^n)$ in a complexity less than 2^κ. ▷

The fact that a block cipher can be distinguished is not necessarily a security flaw. However, when a distinguishing property, which is often called a *distinguisher*, is discovered, it is likely that confidentiality issues will follow. Furthermore, it clearly establishes the fact that the hypothesis made to prove the security of modes of operation is not satisfied in that specific case. With the variety and the number of existing block ciphers, if a cipher is distinguishable, it should already be considered broken and not be used anymore.

In order to study the resistance against distinguishing attacks, there exist today many standard techniques. From a general perspective, these techniques all leverage some weaknesses of the round function (or of one of its components) and try to expand it to more rounds. The major shield against such generalizations to more rounds is the addition of distinct, and ideally, independent round constants. However, some techniques can somehow "bypass" key additions. This is the case for instance of *differential* and *linear* cryptanalysis.

Differential cryptanalysis. Differential cryptanalysis [BS91b, BS91a] is interested in measuring how much closely-related plaintexts can lead to closely-related ciphertexts. Ideally, the encryption of two close messages should be very different. In the opposite case, this could be a serious confidentiality threat. To do so, the cryptanalyst starts with pairs $(x, x+\Delta^{\text{in}})$ of plaintexts, where x is drawn uniformly at random, and where Δ^{in} is the *difference* between the two inputs, because we work with modulo 2 addition. Then, the successive differences taken by the partial encryption of x and $x+\Delta^{\text{in}}$ throughout the block cipher are tracked. In the end, a distinguisher is found when we can establish the fact that the ciphertexts will differ with a difference Δ^{out} with a higher probability (taken over the values $x \in \{0,1\}^n$) than for a random function. This is made easier by the fact that after a constant addition an input pair $(x, x+\Delta)$ becomes $(x+c, x+\Delta+c)$, but still differs with the same difference Δ. Differential cryptanalysis is more precisely introduced in Section 2.3.3.

Linear cryptanalysis. Linear cryptanalysis [Mat94, TG92] is instead interested in finding good linear approximations of the cipher. More precisely, to mount a linear attack, an adversary must be able two find two linear combinations $(a_0, \dots, a_{n-1})$ and $(b_0, \dots, b_{n-1})$ such that $\sum_{i=0}^{n-1} a_i x_i \approx \sum_{i=0}^{n-1} b_i y_i$ holds for many pairs (x, y) where x_i are the bits of plaintext and y_i the bits of ciphertext. If such an equation holds for many inputs, the adversary is able to distinguish the cipher from a random bijection. Again, such an approximation is often found by propagating a linear combination throughout the cipher. This is simplified by the fact that through a constant addition, we obtain $\sum_{i=0}^{n-1} a_i(x_i + c_i) = \sum_{i=0}^{n-1} a_i x_i + \sum_{i=0}^{n-1} a_i c_i$, so that the same linear combination is applied to the input, to the output and to the constant. Linear cryptanalysis is presented in Section 2.3.4.

Algebraic attacks. There exist many other ways to distinguish a cipher from a random bijection. In particular, a lot of attack tackle this problem from an algebraic perspective. Such attacks use the fact that the algebraic representation of the round function is not random-looking. They then manage to either recover equations in the key bits, which then reduce the size of the key space, or point out that some equations take the value 0 (or 1) more often than it is expected for a random bijection. Linearization attacks and interpolation attacks are presented in Sections 2.3.1 and 2.3.2, while the already-mentioned cube attacks and the related higher-order differential attacks are addressed in detail in Chapter 3.

1.5 Problem statement and contributions

In light of this general context, the objective of this thesis is to question the following problem:

> How can the strengths or weaknesses of symmetric ciphers be captured by their algebraic properties?

In order to provide some parts of answer to this arguably open question, this manuscript is composed as follows.

Chapter 2. The second chapter is a general introduction to Boolean (vectorial) functions and their cryptographic analysis. It allows us to introduce all the definitions and notation used throughout this manuscript. It also presents in more detail the framework of differential, linear and algebraic cryptanalysis.

Chapter 3. The third chapter is the first contribution *per se.* After describing in detail the context of higher-order differential cryptanalysis, we apply this methodology to mount an inner-state recovery against Ascon, when the nonce is misused.

Chapter 4. In the fourth chapter, we start by presenting a lightweight cipher called Midori, which is heavily-based on the AES. Next, we review the main *invariant attacks* against it. This is the starting point of our differential analysis of Midori involving non-linear change of variables. Finally, we discuss the relationship between this methodology and the ones of *commutative cryptanalysis* and differential cryptanalysis using *alternative group laws.*

Chapter 5. In the fifth chapter, we continue the study of Midori. More precisely, we develop the framework of commutative cryptanalysis introduced in the previous chapter, and investigate its link with previous attacks based on self-similarities. We also explain how commutative cryptanalysis enables to study the properties highlighted in Chapter 4, by only focusing on the original representation of the cipher. As a side effect, we point out astonishing differential properties of Midori and Scream.

Chapter 6. While the three previous chapters are dedicated to cryptanalysis, the sixth chapter is focused on the theoretical study of some Boolean functions. In particular, we look at generalizations of monomials functions, namely *cyclotomic mappings* and (q, q')-biprojective mappings. We highlight the fact that these notions are very particular cases of linear self-equivalences, and prove that almost all known infinite APN families fall under one of these categories. More generally, we study the properties of linearly self-equivalent mappings, and question the existence of such functions in all CCZ-equivalence class of APN functions.

Chapter 7. Finally, the last chapter is dedicated to the design of primitives for emerging usages. The first half presents two highly-efficient MAC constructions based on the AES. They rely on the extremely fast AES-NI set of instructions available on modern processors. The second part of this chapter focuses on the design of a stream cipher over the field with 17 elements. The objective of this construction is to provide a fast transciphering method to be used in TFHE, which is a specific framework of fully homomorphic encryption.

This manuscript is based on the following collaborative works.

1.6 Publications

Journal

[Bar+24] Augustin Bariant, Jules Baudrin, Gaëtan Leurent, Clara Pernot, Léo Perrin, and Thomas Peyrin. "Fast AES-Based Universal Hash Functions and MACs: featuring LeMac and PetitMac". In: *IACR Transactions on Symmetric Cryptology* 2024.2 (2024), 35–67.

[Bau+23] Jules Baudrin, Patrick Felke, Gregor Leander, Patrick Neumann, Léo Perrin, and Lukas Stennes. "Commutative Cryptanalysis Made Practical". In: *IACR Transactions on Symmetric Cryptology* 2023.4 (2023), pp. 299–329.

[BCP22] Jules Baudrin, Anne Canteaut, and Léo Perrin. "Practical Cube Attack against Nonce-Misused Ascon". In: *IACR Transactions on Symmetric Cryptology* 2022.4 (2022), pp. 120–144.

International conference

[BCP24b] Jules Baudrin, Anne Canteaut, and Léo Perrin. "On functions of $\mathbb{F}_{2^{2t}}$ mapping cosets of $\mathbb{F}_{2^t}^*$ to cosets of $\mathbb{F}_{2^t}^*$". In: *WCC 2024: The Thirteenth International Workshop on Coding and Cryptography*. https://wcc2024.sites.dmi.unipg.it/. Perugia, Italy, 2024, pp. 45–57.

Under submission

[Bau+24a] Jules Baudrin, Christof Beierle, Patrick Felke, Gregor Leander, Patrick Neumann, Léo Perrin, and Lukas Stennes. "Commutative Cryptanalysis as a Generalization of Differential Cryptanalysis". Under submission. 2024.

[Bau+24b] Jules Baudrin, Sonia Belaïd, Nicolas Bon, Christina Boura, Anne Canteaut, Gaëtan Leurent, Pascal Paillier, Léo Perrin, Matthieu Rivain, Yann Rotella, and Samuel Tap. "Transitor: a TFHE-friendly Stream Cipher". Under submission. 2024.

[BCP24a] Jules Baudrin, Anne Canteaut, and Léo Perrin. "Linear self-equivalence of the known families of APN functions: a unified point of view". Under submission. 2024.

Chapter 2

Algebraic tools for Boolean functions analysis

The purpose of this chapter is to establish the main mathematical tools needed throughout the rest of the manuscript. First, the main definitions about finite fields are given. The book by Lidl & Niederreiter [LN96] is our primary reference on this subject. Next, Boolean functions and their different representations are described. In that case, we often refer to the book by Carlet [Car21] which deals with many subjects related to the theoretical study of these objects. Afterwards, the cryptographic criteria associated to Boolean functions are presented. This topic is for instance detailed in the lecture notes by Canteaut [Can16], and we sometimes rely on them. Finally, the main equivalence relations on vectorial Boolean functions that preserve the previous criteria are introduced.

Contents

2.1 The finite field $\mathbb{F}_{2^n}$

As already mentioned in Section 1.3.1, all the functions that are analyzed in this work are functions whose domains and codomains are not only finite, but above all of the form $\{0,1\}^n$. Starting from the finite field with two elements, that is, $\{0,1\}$ with addition and multiplication modulo 2, we can equip $\{0,1\}^n$ with different laws.

The finite field with two elements is denoted by $\mathbb{F}_2$. From there, vectors of $\{0,1\}^n$ can naturally be identified with polynomials of $\mathbb{F}_2[X]$ of degree less than or equal to $n-1$, but also with integers ranging from 0 to 2^n-1:

$$(a_0, \dots, a_{n-1}) \quad \simeq \quad \sum_{i=0}^{n-1} a_i X^i \quad \simeq \quad \sum_{i=0}^{n-1} a_i 2^i. \tag{2.1}$$

Remark 2.1. We use the standard notation where the first coordinate of a vector is located at the leftmost position. Contrary to what the left-to-right writing might suggest, this first coordinate is paired with the least significant bit (LSB) of the associated integer, and *not* with the most significant one (MSB). ▷

In particular, the coordinate-wise addition, or equivalently the polynomial addition or the bitwise one, is a group law for $\mathbb{F}_2^n$. Furthermore, we can consider a scalar multiplication $\cdot\colon \mathbb{F}_2 \times \mathbb{F}_2^n \to \mathbb{F}_2^n$. Because, by definition, it must satisfy for any $x \in \mathbb{F}_2^n$, both $0 \cdot x = 0$ and $1 \cdot x = x$, only a single scalar multiplication exists in that case and it satisfies all the necessary axioms making $(\mathbb{F}_2^n, +, \cdot)$ a $\mathbb{F}_2$-vector space.

Then, by choosing an irreducible polynomial P of degree n, we can further identify $\{0,1\}^n$ with $\mathbb{F}_2[X]/(P)$. This way, $\{0,1\}^n$ is associated to a quotient ring, where the addition is computed coordinate-wise (in $\mathbb{F}_2$), while the multiplication is computed modulo P. Because P is chosen to be irreducible, the obtained ring is actually a field.

Proposition 2.2 (Construction of the finite field with 2^n elements)**.** *Let $n \geq 1$ be an integer. Let $P \in \mathbb{F}_2[X]$ be an irreducible polynomial of degree n. Then $\mathbb{F}_2[X]/(P)$ is a field with 2^n elements.*

Proof. We only prove the existence of an inverse for any non-zero element. Let $Q \in \mathbb{F}_2[X]$ be of degree less than or equal to $n-1$. Because P is irreducible, we know that $\gcd(P, Q) = 1$. But $\mathbb{F}_2[X]$ is an Euclidean domain, so there exists $U, V \in \mathbb{F}_2[X]$ such that $UP + BQ = \gcd(P, Q) = 1$. By looking at this last equality modulo P, we obtain $BQ \equiv 1 \bmod P$, which implies that B is the inverse of Q modulo P. □

Therefore, $\{0,1\}^n$ can be equipped with a field structure. From now on, any field with 2^n elements is denoted by $\mathbb{F}_{2^n}$. This notation is unambiguous because any two fields with 2^n elements are isomorphic, see for instance [LN96, Theorem 2.5].

For the same reason explained above, any field with 2^n elements can naturally be equipped with the trivial scalar multiplication in $\mathbb{F}_2$, making it a $\mathbb{F}_2$-vector space. In the following, by *identifying* $\mathbb{F}_2^n \simeq \mathbb{F}_{2^n}$, we always refer to a vector space isomorphism, that is, to a $\mathbb{F}_2$-linear bijection between both vector spaces. This is naturally equivalent to the choice of a basis of $\mathbb{F}_2^n$ and of a basis of $\mathbb{F}_{2^n}$, and to the identification of both.

The *canonical basis* of $\mathbb{F}_2^n$ is the basis made of the unit vectors $\xi^{(i)}$, for any $i \in [\![0, n-1]\!]$ which are defined by:

$$\forall i \in [\![0, n-1]\!]\,, \quad \xi^{(i)} := (0, \dots, 0, \underset{i\text{th coor.}}{1}, 0, \dots, 0). \tag{2.2}$$

In general, for identifications $\mathbb{F}_2^n \simeq \mathbb{F}_{2^n}$, we *prefer* the usage of Eq. (2.1) and Proposition 2.2, together with the choice of P. This corresponds to the identification of the canonical basis of $\mathbb{F}_2^n$, with the basis $(1, \alpha, \dots, \alpha^{n-1})$ where $\alpha := \bar{X}$ is the class of X in $\mathbb{F}_2[X]/(P)$. When the identification is made the other way around, that is, $\mathbb{F}_{2^n} \simeq \mathbb{F}_2^n$, it is derived from the choice of an element $\alpha \in \mathbb{F}_2^n$ such that $(1, \alpha, \dots, \alpha^{n-1})$ is an $\mathbb{F}_2$-basis of $\mathbb{F}_{2^n}$. These *preferred* identifications are dictated by implementation, as Proposition 2.2 is the most natural way of implementing a finite field on a computer.

Remark 2.3. In the light of the previous observations, we then recommend caution in interpreting the notation $\mathbb{F}_{2^n}$. While mathematically unambiguous (see for instance [LN96, Theorem 2.5]), when *implementing* a function $F\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^m}$, an implementer is free to choose any representation of the fields, or any identifications $\mathbb{F}_{2^n} \simeq \mathbb{F}_2^n$ and $\mathbb{F}_{2^m} \simeq \mathbb{F}_2^m$. These *non-canonical* choices can lead to some non-trivial errors and should be *explicitly identified.* ▷

2.2 Boolean functions

Definition 2.4 (Boolean function)**.** Let $n, m \geq 1$. An (n, m)-*vectorial Boolean function* is a function of the form $F \colon \mathbb{F}_2^n \to \mathbb{F}_2^m$. The name *Boolean function* is reserved to the case $m = 1$. If $m > 1$, the functions $F_0, \ldots, F_{m-1} \colon \mathbb{F}_2^n \to \mathbb{F}_2$ such that $F(x) = (F_0(x), \ldots, F_{m-1}(x))$ for any $x \in \mathbb{F}_2^n$ are called the *coordinates* of F. The number of variables n is sometimes called the *input size*, and the number of coordinates m, the *output size*. ▷

Whenever possible, we reserve upper-case letters to vectorial Boolean functions and lower-case ones to Boolean functions. The coordinate F_i of a vectorial Boolean function F is an exception to the rule.

As for any function, the most natural way to describe a vectorial Boolean function is by pairing each element $x \in \mathbb{F}_2^n$ to its value $F(x) \in \mathbb{F}_2^m$. The sequence $(x, F(x))_{x \in \mathbb{F}_2^n}$ is known as the *look-up table* (shortened *LUT*) of F.

Example 2.5 (Look-up table of the Sbox of Ascon)**.** The AEAD (see Section 1.3.1.b) encryption scheme Ascon uses as internal component the 5-bit Sbox whose look-up table is given in hexadecimal form in Table 2.1. For instance, the fact that the image of `0x2` is `0x1f` can be rewritten using Eq. (2.1) as $S(0, 1, 0, 0, 0) = (1, 1, 1, 1, 1)$. This equivalently means that $S_i(0, 1, 0, 0, 0) = 1$ for any $i \in [\![0, 4]\!]$. ▷

x	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
$S(x)$	4	b	1f	14	1a	15	9	2	1b	5	8	12	1d	3	6	1c
x	10	11	12	13	14	15	16	17	18	19	1a	1b	1c	1d	1e	1f
$S(x)$	1e	13	7	e	0	d	11	18	10	c	1	19	16	a	f	17

Table 2.1: Look-up table of the Sbox of Ascon in hexadecimal notation.

There however exist other representations of vectorial Boolean functions which are better suited in many situations.

2.2.1 Polynomial representations of Boolean functions

Because of the finite field structure, any function is a polynomial function, as detailed in the next proposition.

Proposition 2.6 (Interpolation of a function over finite fields)**.** *Let $n, d \geq 1$, $q = 2^d$ and $f \colon (\mathbb{F}_q)^n \to \mathbb{F}_q$. Then there exists a unique polynomial $P \in \mathbb{F}_q[X_0, \ldots, X_{n-1}]/(X_0^q + X_0, \ldots, X_{n-1}^q + X_{n-1})$ which satisfies*

$$f(x_0, \ldots, x_{n-1}) = P(x_0, \ldots, x_{n-1}) \quad \forall\ x_0, \ldots, x_{n-1} \in \mathbb{F}_q.$$

Proof. Let $i \in [\![0, n-1]\!]$ and $y \in \mathbb{F}_q$. Let us define the Lagrange interpolant

$$P_{i,y} := \prod_{x \in \mathbb{F}_q \setminus \{y\}} \frac{X_i + x}{y + x},$$

which is of degree $|\mathbb{F}_q \setminus \{y\}| = q-1$. We observe that $P_{i,y}(x) = 0$ for all $x \in \mathbb{F}_q \setminus \{y\}$ and $P_{i,y}(y) = 1$. Thus for any $y := (y_0, \ldots, y_{n-1}) \in (\mathbb{F}_q)^n$ and any $\alpha \in \mathbb{F}_q$, the evaluations of the polynomial defined by:

$$Q_{y,\alpha} := \alpha \prod_{i=0}^{n-1} P_{i,y_i}$$

are zero everywhere except when $(x_0, \ldots, x_{n-1}) = (y_0, \ldots, y_{n-1})$, where its value is α. Thus any function $f\colon (\mathbb{F}_q)^n \to \mathbb{F}_q$ can be interpolated using $\sum_{y \in (\mathbb{F}_q)^n} Q_{y,f(y)}$, whose degree in each variable is less than or equal to $q-1$. Moreover, there exist q^{q^n} functions from $\mathbb{F}_q^n$ to $\mathbb{F}_q$ and as many polynomials of $\mathbb{F}_q[X_0, \ldots, X_{n-1}]$ with the announced bound on the degrees, which proves the uniqueness of such a representation. □

This interpolation result gives two distinct and fruitful representations of vectorial Boolean functions. The first one is obtained by applying Proposition 2.6 to each coordinate of a vectorial Boolean function.

Definition 2.7 (Algebraic normal form (ANF))**.** Let $n, m \geq 1$. Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$. The *algebraic normal form* (ANF) of F is the family of interpolation polynomials (given by Proposition 2.6 for $d = 1$) $P_0, \ldots, P_{m-1} \in \mathbb{F}_2[X_0, \ldots, X_{n-1}]/(X_0^2 + X_0, \ldots, X_{n-1}^2 + X_{n-1})$ of the coordinate functions of F. ▷

This representation is multivariate, as any coordinate is expressed as a polynomial in multiple variables. However, when m is a divisor of n, that is when $\mathbb{F}_{2^m} \subset \mathbb{F}_{2^n}$, F can be seen as a function $F\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^m} \subset \mathbb{F}_{2^n}$. In that case, a univariate representation of F can also be considered.

Definition 2.8 (Univariate representation)**.** Let $n \geq 1$ and $F\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$. The *univariate representation* of F is the interpolation polynomial $P \in \mathbb{F}_{2^n}[X]/(X^{2^n} + X)$ of F that is given by Proposition 2.6. ▷

It is possible to refer to *the* univariate representation of a vectorial Boolean function. This however implies that the same identification $\mathbb{F}_{2^n} \simeq \mathbb{F}_2^n$ is used for both the domain and codomain, and that it is *explicit* and clear from context.

Example 2.9 (Multivariate and univariate representations of the Sbox of Ascon)**.** The Sbox of Ascon, that is presented in Example 2.5, has the following ANF:

$$\begin{aligned} S\colon (\mathbb{F}_2)^5 &\to (\mathbb{F}_2)^5 \\ \begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} &\mapsto \begin{pmatrix} x_0x_3 + x_0 + x_1 + x_3x_4 + x_3 \\ x_0x_4 + x_0 + x_1x_4 + x_1 + x_2 + x_3 + x_4 \\ x_0x_1 + x_0 + x_2 + x_3 + 1 \\ x_0 + x_1x_2 + x_1x_3 + x_1 + x_2x_3 + x_2 + x_3 + x_4 \\ x_0x_3 + x_1 + x_2x_3 + x_2 + x_3x_4 + x_3 + x_4 \end{pmatrix}. \end{aligned} \tag{2.3}$$

Moreover by using $\mathbb{F}_2[Y]/(P) \simeq \mathbb{F}_2^5$ where $P = Y^5 + Y^2 + 1$ and by denoting $a \equiv Y \bmod P$, we obtain the following univariate representation of S:

$$S = a^{24}X^{24} + a^4X^{20} + a^{16}X^{18} + a^{27}X^{17} + a^{17}X^{16} + a^{28}X^{12} + a^7X^{10} +$$
$$a^{22}X^9 + a^{16}X^8 + a^{16}X^6 + a^{22}X^5 + a^{17}X^4 + X^3 + a^{12}X^2 + a^{15}X + a^2.$$

▷

From now on, we no longer make a distinction between a function and the polynomials associated to it, and always refer to the reduced forms presented in Definitions 2.7 and 2.8.

Regarding notation, if $u = (u_0, \ldots, u_{n-1}) \in \mathbb{F}_2^n$, the monomial $\prod_{i=0}^{n-1} X_i^{u_i}$ is denoted by X^u. If $x = (x_0, \ldots, x_{n-1}) \in \mathbb{F}_2^n$, the same applies for the value $\prod_{i=0}^{n-1} x_i^{u_i}$ which is denoted by x^u. For any $u, v \in \mathbb{F}_2^n$, we denote by $\preceq$ the *covering relation*:

$$u \preceq v \iff \mathrm{Supp}(u) \subset \mathrm{Supp}(v).$$

While the construction in the proof of Proposition 2.6 can be used in practice, it does not *explicitly* provide the value of the coefficients of the interpolation polynomial. In the case of the ANF, they can be efficiently exhibited thanks to the well-known coefficients-values relations.[1]

Proposition 2.10 (Coefficients-values relations). *Let $f\colon \mathbb{F}_2^n \to \mathbb{F}_2$ be defined by $f(x) = \sum_{u \in \mathbb{F}_2^n} a_u x^u$, where $a_u \in \mathbb{F}_2$ for any $u \in \mathbb{F}_2^n$. Then:*

$$\forall\ u \in \mathbb{F}_2^n, \quad a_u = \sum_{v \preceq u} f(v) \quad \textit{and} \quad f(u) = \sum_{v \preceq u} a_v.$$

Proof. Adapted from [Can16, Theorem 1.3]. Let $u, v \in \mathbb{F}_2^n$. First of all, we observe that $v^u = 1$ if and only if $v_i^{u_i} = 1$ for any $i \in [\![0, n-1]\!]$. This can equivalently be restated as: for any i, if $u_i = 1$, then $v_i = 1$. Equivalently, $v^u = 1$ if and only if $\mathrm{Supp}(u) \subset \mathrm{Supp}(v)$. Thus, for any $v \in \mathbb{F}_2^n$, $f(v) = \sum_{u \in \mathbb{F}_2^n} a_u v^u = \sum_{u \preceq v} a_u$. Conversely, for a fixed $u \in \mathbb{F}_2^n$, we observe that:

$$\sum_{v \preceq u} f(v) = \sum_{v \preceq u} \sum_{w \preceq v} a_w = \sum_{w \preceq u} \sum_{w \preceq v \preceq u} a_w = \sum_{w \preceq u} 2^{\dim(\mathrm{Supp}(u)) - \dim(\mathrm{Supp}(w))} a_w = a_u,$$

where the last equality holds because $2^{\dim(\mathrm{Supp}(u)) - \dim(\mathrm{Supp}(w))}$ is even (and therefore equals 0 mod 2) whenever $w \neq u$. □

[1] In the case of a univariate polynomial $P \in \mathbb{F}_{2^n}[X]$, such relations between coefficients and values of P also exist through Fourier transform, but are not used in this work.

2.2.2 The Walsh transform

2.2.2.a The case of Boolean functions

For any $u \in \mathbb{F}_2^n$, let us denote by $\mathbf{1}_u \colon \mathbb{F}_2^n \to \mathbb{F}_2$ the indicator function of $\{u\}$ whose value is 0 everywhere except for $x = u$ where its value is 1. The LUT of a Boolean function $f \colon \mathbb{F}_2^n \to \mathbb{F}_2$ actually corresponds to the decomposition of f as a linear combination of these indicator functions:

$$f = \sum_{u \in \mathbb{F}_2^n} f(u)\mathbf{1}_u.$$

The indicator functions constitute the *standard* basis of the *complex-valued Boolean function* space: $\mathbb{C}^{\mathbb{F}_2^n} := \{f \colon \mathbb{F}_2^n \to \mathbb{C}\}$. However this basis is not well-suited for studying the behavior of a Boolean function with respect to the group structure of $\mathbb{F}_2^n$: it is more convenient to work with the *characters* basis. This basis is made of the group homomorphisms $\chi \colon \mathbb{F}_2^n \to \mathbb{C}^*$ which, by definition, carry the group structure of $\mathbb{F}_2^n$ into $\mathbb{C}^*$. The change-of-basis operation from the standard basis to the character basis is the well-known Fourier transform.

Definition 2.11 (Character)**.** For any $u \in \mathbb{F}_2^n$, the *character* associated to u is the function denoted by $\chi_u \colon \mathbb{F}_2^n \to \mathbb{C}$ and defined by:

$$\forall x \in \mathbb{F}_2^n, \quad \chi_u(x) := (-1)^{u \cdot x},$$

where the dot $\cdot$ corresponds to the standard dot product that is defined by:

$$\forall\ x, y \in \mathbb{F}_2^n, \quad x \cdot y := \sum_{i=0}^{n-1} x_i y_i.$$

▷

Lemma 2.12 (Mean of a character)**.** *Let $u \in \mathbb{F}_2^n$. Then*

$$\sum_{v \in \mathbb{F}_2^n} \chi_u(v) = \begin{cases} 2^n & \text{if } u = 0 \\ 0 & \text{otherwise,} \end{cases} \tag{2.4}$$

or equivalently, $\sum_{v \in \mathbb{F}_2^n} \chi_u(v) = 2^n \mathbf{1}_0(u)$.

Proof. If $u = 0$, the result is clear. Otherwise, there exists $w \in \mathbb{F}_2^n$ such that $\chi_u(w) = -1$. Thus,

$$-\sum_{v \in \mathbb{F}_2^n} \chi_u(v) = \chi_u(w) \sum_{v \in \mathbb{F}_2^n} \chi_u(v) = \sum_{v \in \mathbb{F}_2^n} \chi_u(v + w) = \sum_{v \in \mathbb{F}_2^n} \chi_u(v),$$

so we necessarily get $\sum_{v \in \mathbb{F}_2^n} \chi_u(v) = 0$. □

Proposition 2.13 (Fourier transform)**.** *Let $n \geq 1$. The family $(\chi_u)_{u\in\mathbb{F}_2^n}$ is a basis of $\mathbb{C}^{\mathbb{F}_2^n}$. The* Fourier transform *of $f\colon \mathbb{F}_2^n \to \mathbb{C}$ is defined by the function $\hat{f}\colon \mathbb{F}_2^n \to \mathbb{C}$ which corresponds to the decomposition of f in the character basis:*

$$f =: \sum_{u\in\mathbb{F}_2^n} \hat{f}(u)\chi_u. \tag{2.5}$$

Proof. The cardinality of $(\chi_u)_{u\in\mathbb{F}_2^n}$ is the same as the one of the standard basis. It is therefore sufficient to show that $(\chi_u)_{u\in\mathbb{F}_2^n}$ is a linearly independent family. Let $f = \sum_{u\in\mathbb{F}_2^n} a_u\chi_u = 0$. Let $v \in \mathbb{F}_2^n$. Then

$$0 = \sum_{u\in\mathbb{F}_2^n} f(u)\chi_v(u) = \sum_{w\in\mathbb{F}_2^n} a_w \sum_{u\in\mathbb{F}_2^n} \chi_v(u)\chi_w(u) = \sum_{w\in\mathbb{F}_2^n} a_w \sum_{u\in\mathbb{F}_2^n} \chi_{v+w}(u) = 2^n a_v,$$

where the last equality comes from the mean of χ_{v+w} (see Lemma 2.12). We finally obtain $a_v = 0$ for any $v \in \mathbb{F}_2^n$. □

By evaluating Eq. (2.5) at each point $v \in \mathbb{F}_2^n$, we easily switch from the character basis to the standard basis. The next proposition enables the change-of-basis the other way around.

Proposition 2.14 (Fourier inversion)**.** *Let $f\colon \mathbb{F}_2^n \to \mathbb{C}$. Then $\hat{f}$ is defined by:*

$$\hat{f} = 2^{-n} \sum_{u\in\mathbb{F}_2^n} f(u)\chi_u.$$

Proof. Let $v \in \mathbb{F}_2^n$. By substituting $f(u)$ using Eq. (2.5) and by following the exact same path as the proof of Proposition 2.13 we observe that:

$$\sum_{u\in\mathbb{F}_2^n} f(u)\chi_u(v) = \sum_{w\in\mathbb{F}_2^n} \hat{f}(w) \sum_{u\in\mathbb{F}_2^n} \chi_u(v)\chi_w(u) = \sum_{w\in\mathbb{F}_2^n} \hat{f}(w) \sum_{u\in\mathbb{F}_2^n} \chi_{v+w}(u) = 2^n\hat{f}(v).$$

□

In the character basis, all the linear operators $f \mapsto f(\cdot + t)$ are diagonal operators. Indeed denoting $g = f(\cdot + t)$, we easily verify that $\hat{g} = \chi_t\hat{f}$. This enables an easier study of the effect of affine shifts on a function. For a Boolean function $f\colon \mathbb{F}_2^n \to \mathbb{F}_2$, the study of the closely-related function $(-1)^f$ from $\mathbb{F}_2^n$ to $\{-1, 1\}$ that is defined by $x \mapsto (-1)^{f(x)}$, and in particular of its Fourier transform, is exceptionally fruitful to understand how f behaves with respect to the group law, that is, the addition in $\mathbb{F}_2^n$.

Definition 2.15 (Walsh transform)**.** Let $f\colon \mathbb{F}_2^n \to \mathbb{F}_2$. The Walsh transform of f, that is denoted by $W_f\colon \mathbb{F}_2^n \to \mathbb{Z}$, is (up to a factor 2^n) the Fourier transform of $(-1)^f$. More precisely, it is defined by:

$$\forall u \in \mathbb{F}_2^n, \quad W_f(u) := \sum_{v\in\mathbb{F}_2^n} (-1)^{f(v)+u\cdot v}. \tag{2.6}$$

▷

The Walsh transform is indeed a *transform* because the Fourier transform, the transform $f \mapsto (-1)^f$, and the scaling by 2^n are all bijective. From Eq. (2.6), we observe that the Walsh transform captures the signed distance of a function to all the linear functions $x \mapsto u \cdot x$.

2.2.2.b The case of vectorial Boolean functions

Regarding a vectorial Boolean function $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$, the knowledge of the Walsh transform of each of its coordinates is equivalent to the knowledge of F itself. However in general, we rather work with the Walsh coefficients of *all* linear combinations of coordinates of F. Let $\beta \in \mathbb{F}_2^m$. We denote by $\beta \cdot F$ the Boolean function $x \mapsto \sum_{i=0}^{m-1} \beta_i \cdot F_i(x)$. Such a combination is called a *component* of F. Furthermore, for any $\alpha \in \mathbb{F}_2^n$ we denote by $W_F(\alpha, \beta)$ the Walsh coefficient of $\beta \cdot F$ associated to α:

$$W_F(\alpha, \beta) := W_{\beta \cdot F}(\alpha) = \sum_{x \in \mathbb{F}_2^n} (-1)^{\alpha \cdot x + \beta \cdot F(x)}.$$

When considering a coefficient $W_F(\alpha, \beta)$, we refer to α as the *input mask* and to β as the *output mask*. The matrix $(W_F(\alpha, \beta))_{\alpha \in \mathbb{F}_2^n, \beta \in \mathbb{F}_2^m}$ is known as the *linear approximation table* (LAT). It is sometimes more convenient to work with the coefficients scaled up to 2^{-n} factor. They are denoted by:

$$\forall \alpha \in \mathbb{F}_2^n, \forall \beta \in \mathbb{F}_2^m, \quad \widetilde{W}_F(\alpha, \beta) := \frac{1}{2^n} W_F(\alpha, \beta), \tag{2.7}$$

and the matrix $\mathbf{C}^F := \left(\widetilde{W}_F(\alpha, \beta)\right)_{\alpha \in \mathbb{F}_2^n, \beta \in \mathbb{F}_2^m}$ is known as the *correlation matrix* [DGV95] of F.

Storing information about all components, rather than just all coordinates, is redundant but the LAT has the advantage of being very convenient to use. Indeed, let us denote by $\mathbf{T}^F : \mathbb{C}^{\mathbb{F}_2^n} \to \mathbb{C}^{\mathbb{F}_{2^m}}$ the linear operator called *pushforward operator* of F. This operator $\mathbf{T}^F$ is defined [Bey21, Definition 3.2] with respect to the standard basis by:

$$\forall\ x \in \mathbb{F}_2^n, \quad \mathbf{T}^F(\mathbf{1}_x) := \mathbf{1}_{F(x)}.$$

In the character basis, the matrix of $\mathbf{T}^F$ becomes $\mathbf{C}^F = \frac{1}{2^n}\mathrm{LAT}(F)$, see [Bey21, Definition 3.3]. The LAT therefore inherits useful properties. As an example, because $\mathbf{T}^{G \circ F} = \mathbf{T}^G \circ \mathbf{T}^F$, the LAT of a composition of functions is the product of the LAT of G and F, up to rescaling.

Remark 2.16. In the standard basis, the matrix of $\mathbf{T}^F$, that is called *transition matrix*, is nothing more than the *adjacency matrix* of the (directed) graph of F. Indeed, each column contains only zeros but a single one. The ones are located at coordinates $(x, F(x))$ for any $x \in \mathbb{F}_2^n$. We therefore observe the redundancy that is mentioned before: the set of coordinates of non-zero entries, which is sufficient to build such a matrix, is exactly the graph of F, or equivalently, its look-up table. ▷

Example 2.17 (LAT of the Sbox of Ascon)**.** The LAT of the Sbox of Ascon is given in Table 2.2, where the input masks are enumerated vertically, and the output ones horizontally. For an easier visualization of such a table, we often rely on a graphical representation instead, as depicted in Figure 2.1.

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f	10	11	12	13	14	15	16	17	18	19	1a	1b	1c	1d	1e	1f
0	32	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
1	·	·	·	·	·	·	16	·	·	8	8	·	·	-8	8	·	·	·	8	8	·	·	8	-8	8	·	-8	·	-8	·	-8	·
2	·	·	·	·	·	·	-16	16	·	·	8	8	·	·	8	8	·	·	8	8	·	·	-8	-8	·	·	·	·	·	·	·	·
3	·	16	·	·	·	·	·	·	·	8	·	8	·	8	·	-8	-16	·	·	·	·	·	·	·	8	·	8	·	8	·	-8	·
4	·	·	·	8	·	-8	·	·	·	·	8	·	·	8	-8	-8	·	·	8	·	-8	·	·	·	·	-16	·	-8	-8	·	8	-8
5	·	·	·	8	·	8	·	·	·	-8	·	·	·	·	·	-8	·	·	·	-8	8	·	-8	-8	8	·	-8	8	·	-16	·	-8
6	·	·	·	8	·	-8	·	·	·	·	·	-8	·	8	·	·	·	·	·	-8	-8	·	-8	-8	·	16	·	-8	-8	·	-8	8
7	·	·	·	-8	·	-8	·	·	·	8	8	8	·	·	-8	·	·	·	-8	·	-8	·	·	·	-8	·	-8	8	·	-16	·	8
8	·	·	·	·	·	·	·	·	·	·	8	8	·	·	-8	-8	·	·	·	·	·	·	·	·	·	16	-8	8	·	16	8	-8
9	·	·	·	·	·	·	·	-16	·	-8	·	8	·	8	·	8	·	·	8	8	·	·	-8	8	8	·	·	8	-8	·	·	8
a	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	8	8	·	·	8	8	·	16	8	-8	·	-16	8	-8
b	·	16	·	·	·	·	·	·	·	-8	8	·	·	-8	-8	·	16	·	·	·	·	·	·	·	8	·	·	-8	8	·	·	8
c	·	·	-16	8	-16	-8	·	·	·	·	·	8	·	-8	·	·	·	·	-8	·	8	·	·	·	·	·	8	·	-8	·	·	·
d	·	·	·	-8	-16	8	·	·	·	8	-8	-8	·	·	-8	·	·	·	·	8	-8	·	-8	-8	8	·	·	·	·	·	8	·
e	·	·	·	-8	16	-8	·	·	·	·	-8	·	·	-8	-8	-8	·	·	·	8	8	·	-8	-8	·	·	8	·	-8	·	·	·
f	·	·	16	-8	-16	-8	·	·	·	-8	·	·	·	·	·	-8	·	·	8	·	8	·	·	·	-8	·	·	·	·	·	-8	·
10	·	·	·	·	·	·	-16	·	·	8	·	-8	-8	·	-8	·	·	·	·	·	8	-8	8	8	8	·	-8	·	-8	·	-8	·
11	·	·	·	·	·	·	·	·	-16	·	-8	8	-8	-8	·	·	·	16	8	-8	-8	-8	·	·	·	·	·	·	·	·	·	·
12	·	-16	·	·	·	·	·	·	·	-8	8	·	-8	·	·	-8	·	·	-8	8	-8	-8	·	·	8	·	8	·	8	·	-8	·
13	·	·	·	·	·	·	-16	-16	·	·	·	·	8	-8	8	-8	·	·	·	·	-8	8	8	-8	·	·	·	·	·	·	·	·
14	·	·	·	8	·	8	·	·	·	8	8	-8	-8	-8	·	-8	·	·	8	·	·	8	-8	8	-8	·	8	8	·	·	·	8
15	·	·	·	8	·	-8	·	·	·	·	·	-8	8	·	-8	8	·	16	·	8	·	8	·	·	·	·	·	8	8	·	-8	-8
16	·	·	·	-8	·	-8	·	·	·	8	·	·	-8	8	8	·	16	·	·	-8	·	8	·	·	8	·	8	8	·	·	·	-8
17	·	·	·	8	·	-8	·	·	16	·	-8	·	-8	·	·	·	·	·	8	·	·	-8	8	-8	·	·	·	8	8	·	8	8
18	·	·	·	·	·	·	·	-16	·	8	8	·	-8	·	·	8	·	·	·	·	8	-8	-8	-8	-8	·	·	-8	8	·	·	-8
19	·	·	·	·	·	·	·	·	·	·	·	·	8	-8	-8	8	·	-16	8	-8	-8	-8	·	·	·	·	8	8	·	·	-8	-8
1a	·	16	·	·	·	·	·	·	·	-8	·	-8	-8	·	8	·	·	·	-8	8	-8	-8	·	·	-8	·	·	8	-8	·	·	-8
1b	·	·	·	·	·	·	·	·	16	·	-8	8	-8	-8	·	·	·	·	·	·	-8	8	-8	8	·	·	-8	-8	·	·	-8	-8
1c	·	·	16	8	·	-8	·	·	·	8	·	·	8	-8	8	·	·	·	-8	·	·	-8	-8	8	8	·	·	·	·	·	8	·
1d	·	·	·	-8	·	8	·	·	16	·	8	·	8	·	·	·	·	16	·	-8	·	-8	·	·	·	·	8	·	-8	·	·	·
1e	·	·	·	8	·	8	·	·	·	8	-8	8	8	8	·	-8	16	·	·	8	·	-8	·	·	-8	·	·	·	·	·	-8	·
1f	·	·	16	8	·	8	·	·	·	·	·	8	-8	·	-8	8	·	·	-8	·	·	8	8	-8	·	·	8	·	-8	·	·	·

Table 2.2: LAT of the Sbox of Ascon. A dot corresponds to a 0 value.

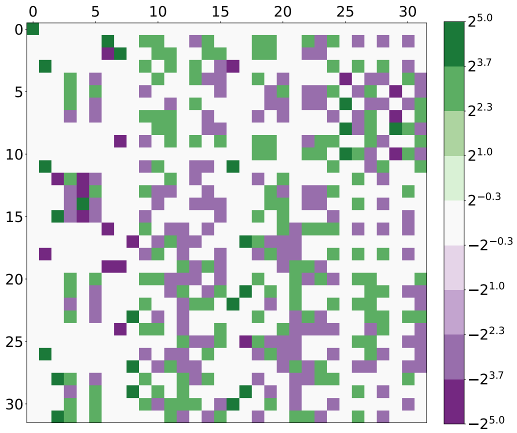


Figure 2.1: LAT of the Sbox of Ascon using two grayscales for positive and negative values.

Equipped with these four representations of vectorial Boolean functions (the look-up table, the univariate polynomial, the ANF and the Walsh transform), a cryptographer has at hand a range of tools to measure the strength or weakness of a function. This methodology is presented in the next section.

2.3 Cryptographic Boolean functions

As already mentioned in Section 1.3.2.b, most of the block ciphers today still heavily rely on the notions of diffusion and confusion introduced by Shannon [Sha49]. The Sboxes, which are the only nonlinear components of the main block cipher constructions, are in charge of confusion. Paraphrasing Shannon, the goal of confusion is to make the relation between the key and the plaintext as intricate as possible.

The worst level of "intrication" is therefore linearity. Indeed, let $k \in \mathbb{F}_{2^\kappa}$, and $E_k \colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ be an instantiated linear key-alternating block cipher. Because of linearity, any plaintext/ciphertext pair (x, y) gives an equation (in $\mathbb{F}_2^n$) of the form $y = E_1(x) + E_2(k)$, where E_1 is linear and bijective and where $E_2(k)$ is linear in the rounds keys $k^{(0)}, \ldots, k^{(R-1)}$. This equation can be rewritten as $E_2(k) = y + E_1(x)$. In other words, from a single plaintext/ciphertext pair, the value $E_2(k)$ is recovered. Let y' be a known ciphertext. Its corresponding plaintext x' can then be recovered by the adversary, as $E_1^{-1}(y' + E_2(k)) = x'$, because E_1 is publicly known and $E_2(k)$ has been previously recovered. The confidentiality of the cipher is then broken by a single known plaintext/cipher pair, *even if the master key has not been recovered.* This is furthermore the case, independently of the number of rounds, and of the circuit complexity of the implementation.

This simple example highlights how much linearity should be avoided. The main cryptographic criteria, that are presented below, all try to capture the distance between a function and the linear functions, using different metrics. In the following, the main definitions used to benchmark cryptographic vectorial Boolean functions, and in particular Sboxes, are presented, together with the necessary tools. The legitimacy of those criteria is illustrated by a (non-exhaustive) selection of known attacks applicable to ciphers which do not meet these requirements.

2.3.1 Algebraic degree and density

2.3.1.a Algebraic degree

We denote by $\mathrm{wt} \colon \mathbb{F}_2^n \to \mathbb{N}$ the Hamming weight, that is, the function defined by $u \mapsto |\mathrm{Supp}(u)|$. The most natural gauge of linearity is inevitably the so-called algebraic degree.

Definition 2.18 (Algebraic degree)**.** Let $f \colon \mathbb{F}_2^n \to \mathbb{F}_2$ be the Boolean function defined by $x \mapsto \sum_{u \in \mathbb{F}_2^n} a_u x^u$, with $a_u \in \mathbb{F}_2$ for any u. The *algebraic degree* of f is the degree of its ANF, *i.e.* $\deg_{\mathrm{a}}(f)$ is defined by:

$$\deg_{\mathrm{a}}(f) := \max(\{\mathrm{wt}(u), u \in \mathbb{F}_2^n, a_u \neq 0\}).$$

The algebraic degree of a vectorial Boolean function $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$ is the maximum degree of its coordinates:

$$\deg_{\mathrm{a}}(F) := \max(\{\deg_{\mathrm{a}}(F_i), i \in \llbracket 0, m-1 \rrbracket\}).$$

▷

A function $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$ whose algebraic degree is exactly 0 (resp. 1, 2) is called a *constant* (resp. *affine*, *quadratic*) function. An affine function satisfying $F(0) = 0$ is called *linear*. This definition of course coincides with the usual definition of $\mathbb{F}_2$-linear functions as functions carrying the addition of $\mathbb{F}_2^n$ toward $\mathbb{F}_2^m$.

While the algebraic degree is the degree of the ANF, it can still be computed using the univariate representation.

Proposition 2.19 (Algebraic degree of a univariate polynomial)**.** *Let $F\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ be defined by $x \mapsto \sum_{j=0}^{2^n-1} a_j x^j$, where $a_j \in \mathbb{F}_{2^n}$ for any j. Then:*

$$\deg_a(F) = \max(\{\mathrm{wt}(j), j \in \llbracket 0, 2^n - 1 \rrbracket, a_j \neq 0\}).$$

Proof. Adapted from [Car21, Proposition 6]. For the sake of this proof, let us introduce for any function F (or polynomial), the quantity $M(F)$ defined by:

$$M(F) := \max(\{\mathrm{wt}(j), j \in \llbracket 0, 2^n - 1 \rrbracket, a_j \neq 0\}).$$

Let $(\alpha_0, \ldots, \alpha_{n-1})$ be an $\mathbb{F}_2$-basis of $\mathbb{F}_{2^n}$ and let $x = \sum_{i=0}^{n-1} x_i \alpha_i \in \mathbb{F}_{2^n}$ where $x_i \in \mathbb{F}_2$ for any $i \in \llbracket 0, n-1 \rrbracket$. The binary decomposition of each integer $j \in \llbracket 0, 2^n - 1 \rrbracket$ is given by $j = \sum_{s=0}^{n-1} j_s 2^s$, where each $j_s \in \{0, 1\}$. Then, $F(x)$ can be decomposed as:

$$F(x) = \sum_{j=0}^{2^n-1} a_j \left(\sum_{i=0}^{n-1} x_i \alpha_i\right)^j = \sum_{j=0}^{2^n-1} a_j \prod_{s=0}^{n-1} \left(\sum_{i=0}^{n-1} x_i \alpha_i^{2^s}\right)^{j_s},$$

where we use the binary decomposition of each j, the linearity of the Frobenius automorphism $y \mapsto y^2$, and the fact that $x_i = x_i^2$ for any i because $x_i \in \mathbb{F}_2$. For any j, it is clear that no product of more than $\mathrm{wt}(j)$ variables $x_0, \ldots, x_{n-1}$ appears in $a_j \prod_{s=0}^{n-1} \left(\sum_{i=0}^{n-1} x_i \alpha_i^{2^s}\right)^{j_s}$. Therefore, it holds that: $\deg_{\mathrm{a}}(F) \leq M(F)$. This means that a polynomial $P = \sum_{j=0}^{2^n-1} a_j X^j$ for which $M(P) \leq d$ for a given $d \in \llbracket 0, n \rrbracket$, defines a function with algebraic degree less than or equal to d. Therefore, the mapping φ_d defined by:

$$\begin{array}{rccc} \varphi_d\colon & \{P \in \mathbb{F}_{2^n}[X]/(X^{2^n} + X), M(P) \leq d\} & \to & \{F\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}, \deg_{\mathrm{a}}(a) \leq d\} \\ & P & \mapsto & x \mapsto P(x) \end{array}$$

is well defined. The mapping φ_d is also injective because of Proposition 2.6. However, there are 2^{nm} elements in its domain where $m = \sum_{i=0}^{d} \binom{n}{i}$, as there are m integers j of at most n bits such that $\mathrm{wt}(j) \leq d$ and for each such j, a_j can be freely chosen in $\mathbb{F}_{2^n}$. But this is also the cardinality of the codomain: F satisfies $\deg_{\mathrm{a}}(F) \leq d$ if and only if $\deg_{\mathrm{a}}(F_i) \leq d$ for each of its n coordinates $F_i\colon \mathbb{F}_2^n \to \mathbb{F}_2$. Each F_i can be freely built by choosing a coefficient in $\mathbb{F}_2$ for each of the m monomials with d variables among n possible ones, which leads to $(2^{2^d})^n$. We therefore conclude that the equality $\deg_{\mathrm{a}}(F) = M(F)$ must hold by following an inclusion-exclusion principle on all values $d \in \llbracket 0, n \rrbracket$. □

2.3.1.b Linearization attack

In general, a cryptographic function should have a *high* algebraic degree. Indeed, if an attacker is given a ciphertext $y = F(x)$, for an unknown function $F \colon \mathbb{F}_2^n \to \mathbb{F}_2^m$ and an unknown $x \in \mathbb{F}_2^n$, he/she can always express any output bit y_i, as a nonlinear equation in unknown variables $x_0, \ldots, x_{n-1} \in \mathbb{F}_2$ and unknown coefficients $a_{i,u} \in \mathbb{F}_2$ for any $u \in \mathbb{F}_2^n, i \in [\![0, m-1]\!]$:

$$\forall i \in [\![0, m-1]\!]\,,\ y_i = \sum_{u \in \mathbb{F}_2^n} a_{i,u} x^u .$$

But if $x \in \mathbb{F}_2^n$ is *known*, then all the monomials $x^u, u \in \mathbb{F}_2^n$ are known, and the nonlinear equations boil down to linear equations in unknowns $a_{i,u} \in \mathbb{F}_2$ for all u and i. Therefore, even if the cryptographic function is nonlinear, Gaussian elimination can always be applied to recover its ANF. We refer to such techniques as *linearization attacks*, see for instance [KS99, BG05, Gil+23]. The time complexity of mounting a linearization attack is therefore $O(M^\omega)$, where $2 \le \omega \le 3$ is the matrix multiplication exponent, and where M is the number of unknowns of the linear system, that is, the number of monomials M. As the number of monomials of degree less than or equal to d in n variables is $\sum_{i=0}^{d} \binom{n}{i}$, the value of M grows exponentially with the degree, and so do the time and data complexities. This is highlighted by Figure 2.2. Data complexity is however linear in M.

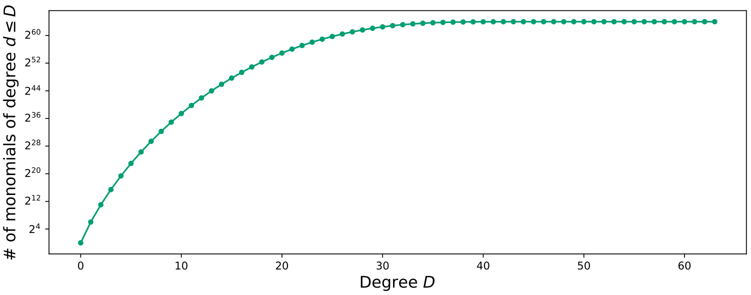


Figure 2.2: Number of monomials in 64 Boolean variables.

The degree therefore gives an upper bound on the security against such techniques, but it is the *precise* number of monomials M that gives the *actual* complexity. Stated otherwise, the *density* or *sparsity* of the ANF is the good measurement of security in that case. Having a dense ANF is necessary to avoid this generic system solving. But it is also desirable to avoid *ad hoc* solving algorithms taking advantages of specificity of the system. The more random-looking is the system, the harder it is to find and leverage a specificity. In practice, the full ANF of a real-life cipher is that large that it cannot even be stored on a computer.

This idea of analyzing a cipher by making sure that breaking it is “at least as [hard] as solving a system of simultaneous equations in a large number of unknowns, of a complex type” is already mentioned in the seminal work of Shannon [Sha49].

2.3.1.c Degree as a distinguishing property

Distinguishing properties can also be exhibited when the degree is low. First of all, as suggested by the following proposition, a permutation cannot reach the maximal algebraic degree.

Proposition 2.20 (Degree of a permutation). *Let $n > 1$ and $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ be a permutation. Then $\deg_a(F) \leq n-1$.*

Proof. From Proposition 2.10, we observe that for any coordinate F_j, $j \in [\![0, n-1]\!]$, the highest monomial, that is $\prod_{i=0}^{n-1} X_i$, has as coefficient $a_j = \sum_{u \in \mathbb{F}_2^n} F_j(u)$. But because F is bijective, each of its coordinates takes the value 0 (and also the value 1) half of the time, that is, 2^{n-1} times. This implies that $a_j = 0$ for any j, and thus the degree of F cannot be equal to n. □

Furthermore, a random function $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ is expected to have *no* term of degree $n-1$ with only probability 2^{-n^2}, as each of the n monomials of degree $n-1$, in each of the n coordinates, has probability $\frac{1}{2}$ of being equal to zero. It is therefore expected that a random bijection has algebraic degree exactly $n-1$, and this holds [Wel69, KP02] even if the probability cannot be computed as easily as for a random function.

A cryptographic permutation should then mimic this behavior, and reach this degree. When the degree is lower than expected, a *higher-order differential* distinguisher can be exhibited. This is detailed in Chapter 3.

2.3.1.d Estimates on the algebraic degree

In order to ensure a high algebraic degree for functions that are iteratively built, the most trivial bound comes from the degree of the composition of functions.

Lemma 2.21 (Trivial degree bound). *Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$, $G\colon \mathbb{F}_2^m \to \mathbb{F}_2^\ell$. Then $\deg_a(G \circ F) \leq \deg_a(G)\deg_a(F)$.*

This bound being very general, it can only be applied as a *rule of thumb*, and a dedicated analysis should be carried when designing or analyzing a cipher. There exist finer generic bounds on the degree of iterated constructions such as the one of Canteaut & Videau [CV02], the one of Boura, Canteaut & De Cannière [BCD11], the one of Boura & Canteaut [BC13], or the one of Cid, Grassi, Gunsing, Lüftenegger, Rechberger & Schofnegger [Cid+22]. Those bounds for instance led to a 18-round [BC11] and a 24-round distinguisher [BCD11] on Keccak-f, based on the study of the degree.

This is actually not the only property that can be leveraged by higher-order differential distinguishers. Indeed, as presented in Chapter 3, such distinguishers can also exploit the sparsity of the ANF of a cryptographic function, and especially the value of some well-chosen coefficients.

2.3.1.e Algebraic degree and density of an Sbox

The fact that an Sbox must be nonlinear obviously corresponds to the criterion $\deg_a(S) > 1$. From there, the design choices are numerous. It seems natural to look for an Sbox which achieves the highest possible degree with a dense ANF. With the expected avalanche effect due to diffusion through the linear layer, this could lead to a very dense ANF for the whole construction in only a few rounds. This is for instance the choice made for the AES Sbox [DR02].

When implementing an Sbox as a look-up table such choices seem legitimate, but may not hold anymore when considering constant-time or memory-optimized implementations. For instance, lightweight cryptography promotes new trade-offs between security, performances and cost. This pushes designers to build ciphers which can be implemented very efficiently. As an example, we can observe in Example 2.9 that the ANF of the Sbox of Ascon is rather sparse and is only quadratic. The main reason of this choice is to facilitate cheap and fast bit-sliced implementation, but also threshold implementations and masking [Dob+21]. This fragile aspect of the Sbox, with respect to algebraic arguments, is compensated by a suitable number of rounds and a good interaction with the linear layer to provide diffusion. This at-first-sight simplicity of the Sbox can however be considered a good property. It indeed enables an easier study and understanding of the algebraic properties of (round-reduced versions of) the cipher. The attack against Ascon that we present in Section 3.4 is heavily based on the quadraticity of the round function and leads to a further understanding of its impact on security, when the attacker is given powerful abilities or when the cipher is not properly used.

2.3.2 Univariate degree and density

The degree of the univariate representation of a function is of course also a measurement of linearity, but this time of $\mathbb{F}_{2^n}$-linearity.

Definition 2.22 (Univariate degree)**.** Let $F\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}, x \mapsto \sum_{i=0}^{2^n-1} a_i x^i$. The *univariate degree* of F is the degree of its univariate polynomial:

$$\deg_u(F) := \max_{i \in [\![0, 2^n-1]\!]} (i, a_i \neq 0).$$

▷

2.3.2.a Linearization, again

In the same way as for the algebraic degree, the degree of a cryptographic function should be high, and its univariate representation should be dense. The first reason comes from the already-mentioned linearization attacks. Let $P = \sum_{i \in I} a_i X^i$, be an unknown polynomial whose non-zero coefficients are located in a *known* subset $I \subset [\![0, 2^n - 1]\!]$. As for the ANF, the knowledge of a plaintext/ciphertext pair $x, y \in \mathbb{F}_{2^n}$ gives one linear equation in the unknown coefficients, as all the power of x are known in $y = \sum_{i \in I} a_i x^i$. Therefore denoting $M := |I|$, we obtain the same complexities as the ones presented in Section 2.3.1. Note that the number of operations that is counted in that case is the number of arithmetic operations in $\mathbb{F}_{2^n}$, and not Boolean operations.

2.3.2.b Interpolation attack

There also exists another class of attacks that recovers an unknown polynomial from its values, but this time, with a complexity depending on the degree. This technique is called *interpolation attack* [JK97] and is based on the Lagrange interpolant, that was already encountered in the proof of Proposition 2.6. It is indeed known that given $d+1$ preimage/image pairs $((x_i, y_i))_{i \in [\![0,d]\!]}$, where $x_i, y_i \in \mathbb{F}_{2^n}$, there exists a unique polynomial $P \in \mathbb{F}_{2^n}[X]$ which satisfies $\deg_{\mathrm{u}}(P) \leq d$ and $P(x_i) = y_i$ for any $i \in [\![0,d]\!]$. This polynomial is given by the following formula:

$$P = \sum_{i=0}^{d} y_i \prod_{j=0, j \neq i}^{d} \frac{X + x_j}{x_i + x_j} = \sum_{i=0}^{d} y_i \frac{Q_i}{Q_i(x_i)}, \tag{2.8}$$

where $Q := \prod_{i=0}^{d} X + x_i$ and $Q_i := \frac{Q}{X + x_i}$ for any i. If a bound d on the degree of an unknown function (for instance an instantiated block cipher E_k) is known, the previous formula therefore enables the attacker to recover the full description of the function, using $d+1$ known input/output pairs (that is, plaintext/ciphertext encrypted with the same key k). The time complexity of interpolation using Eq. (2.8) is $O(d^2)$: first compute Q in $O(d^2)$ operations, then, for each of the d values for i, compute Q_i in $O(d)$, evaluate Q_i in x_i in $O(d)$, and finally compute the linear combination in $O(d)$.

Interpolation can be further sped up using a *divide-and-conquer* algorithm. In that case its cost is $O(\mathcal{M}(d)\log(d))$ where $\mathcal{M}(d)$ is the complexity of the multiplication of two polynomials of $\mathbb{F}_{2^n}[X]$ of degree at most d. When a d-th primitive root exists in $\mathbb{F}_{2^n}$, multiplying polynomials can be done in $\mathcal{M}(d) = O(d\log(d))$ with multiplication based on Fast Fourier Transform (FFT). Otherwise, using a method inspired by Schönhage–Strassen algorithm, the multiplication can be computed in $\mathcal{M}(d) = O(d\log(d)\log(\log(d))$ [CK91]. A detailed (yet French) description of fast interpolation, fast multiplication and their complexities is given in [Bos+17, Chapters I.2 & I.5].

Interpolation can also be adapted to the multivariate case, and therefore used to recover an unknown ANF.

2.3.2.c Univariate degree and density of a round function

Little is known on the actual choice that should be made regarding the univariate description of a round function. The intuition is again that a dense high-degree round function leads to a dense and high-degree construction in only a few rounds. When the Sbox is built as a function $\mathbb{F}_2^n \to \mathbb{F}_2^n$, the corresponding univariate representation often matches these criteria. Indeed, a polynomial of $\mathbb{F}_{2^n}[X]$ with peculiar and distinctive properties can only mean that it interacts oddly with the field structure. Such a strong interaction is therefore not expected to happen when the function is built with the vector space $\mathbb{F}_2^n$ in mind. Regarding the degree, a random bijection is expected to have univariate degree $2^n - 2$ most of the time [KP02].

However, there exist designs that do not follow these guidelines and still achieve good security. The most iconic one is the AES Sbox which is built as $S = A \circ S'$ where $S' \colon \mathbb{F}_{2^8} \to \mathbb{F}_{2^8}, x \mapsto x^{254}$, for a specific identification $\mathbb{F}_2^8 \simeq \mathbb{F}_{2^8}$, and $A \colon \mathbb{F}_2^8 \to \mathbb{F}_2^8$ is an $\mathbb{F}_2$-affine mapping. It therefore relies on a high-degree but sparse bijection. This simple description of S' yields an easier understanding of its structure (it behaves as x^{-1} over $\mathbb{F}_{2^8}^*$ and maps 0 to 0), but also of its security [Nyb94]. However the sparsity and algebraic simplicity of S' seems hard to leverage. Indeed, it is already partially compensated after its composition with A: for $\mathbb{F}_{2^8} = \mathbb{F}_2[X]/(X^8 + X^4 + X^3 + X + 1)$, S is represented as:

$$S = aX^{254} + bX^{253} + cX^{251} + dX^{247} + eX^{239} + X^{223} + fX^{191} + gX^{127} + h,$$

where a, b, c, d, e, f, g, h are specific elements of $\mathbb{F}_{2^8}$.

More recently, new symmetric primitives are needed to run in abstract settings such as multi-party computation or zero-knowledge proof systems, see for instance [Bou23, Chapter 1]. For such usages, primitives are directly implemented using the arithmetic of large finite fields ($\mathbb{F}_q$ where $q \in \{2^{64}, 2^{128}\}$), and the number of multiplications in such a field must be minimized. For instance, the block cipher MiMC [Alb+16] uses $x \mapsto x^3$ over $\mathbb{F}_q$ as sparse low-degree round function. As highlighted by the work of Bouvier, Canteaut & Perrin [BCP23], this univariate simplicity makes a thorough analysis possible and provides precise guarantees on the growth of the algebraic degree, round after round. However, the density of the whole construction is also impacted: in somes cases $\frac{31}{32}$ of the coefficients are necessarily 0 [Bou23, Sec. 5.2]. However, this tighter upper bound differs from the generic one by a constant factor and the use of Gaussian elimination, as presented above, still remains out of reach. It is an open problem to determine whether this peculiar structure can be leveraged in an attack.

2.3.3 Differential uniformity

2.3.3.a Cryptographic context

As already mentioned Section 1.4, differential attacks are among the most popular attacks against block ciphers. This class of attacks was first introduced by Biham & Shamir to cryptanalyze reduced versions of DES [BS91b, BS91a], and was quickly applied to the full DES [BS93]. The basis of such attacks against a cryptographic function $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ is the existence of an *input difference* $\Delta^{\text{in}} \in \mathbb{F}_2^m \setminus \{0\}$ and an *output difference* $\Delta^{\text{out}} \in \mathbb{F}_2^m$ such that the following equation has many solutions $x \in \mathbb{F}_2^n$:

$$F(x + \Delta^{\text{in}}) + F(x) = \Delta^{\text{out}}. \tag{2.9}$$

This property is a distinguishing property compared to the behavior of a random function. Indeed for fixed $\Delta^{\text{in}}, \Delta^{\text{out}}$, if F and x are picked uniformly at random, we expect $F(x + \Delta^{\text{in}}) + F(x)$ to behave as random, and as such, to be equal to Δ^{out} only 1 out of 2^n times (or out of $2^n - 1$ for random *bijections*, as x and $x + \Delta^{\text{in}}$ are distinct so $F(x) + F(x + \Delta^{\text{in}}) \neq 0$). In other words, Eq. (2.9) is expected to have on average a single solution.

In the case of a block cipher $\mathcal{E} = (E_k\colon \mathbb{F}_2^n \xrightarrow{\sim} \mathbb{F}_2^n)_{k \in \mathbb{F}_2^\kappa}$, the same kind of distinguisher can be mounted if there exist $\Delta^{\text{in}}, \Delta^{\text{out}} \in \mathbb{F}_2^n, \Delta^{\text{in}} \neq 0$ such that *for many keys* $k \in \mathbb{F}_2^\kappa$, Eq. (2.10) has many solutions $x \in \mathbb{F}_2^n$:

$$E_k(x + \Delta^{\text{in}}) + E_k(x) = \Delta^{\text{out}}. \tag{2.10}$$

This property enables us to distinguish the draw of a random bijection among the block cipher $F \xleftarrow{\$} \mathcal{E}$ from the random draw among all bijections $F \xleftarrow{\$} \mathrm{Bij}(\mathbb{F}_2^n)$. Indeed, given many chosen plaintexts pairs of the form $(x, x + \Delta^{\text{in}})$, and their corresponding ciphertexts, an attacker can compute $E_k(x+\Delta^{\text{in}})+E_k(x)$ and count the number of times Eq. (2.10) holds or not. Such a distinguisher can often be further leveraged to mount key-recovery attacks. This is depicted for instance in detail in the thesis of Heim [Hei24, Section 2.2].

2.3.3.b Derivatives and differential uniformity

The main objects at hand are the derivatives of vectorial Boolean functions.

Definition 2.23 (Derivative of a function)**.** Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$. Let $\Delta \in \mathbb{F}_2^n$. The *derivative of F along* Δ (or *with respect to* Δ) is the function $D_\Delta(F)\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$ that is defined by:

$$D_\Delta(F)\colon x \mapsto F(x + \Delta) + F(x).$$

▷

Remark 2.24. This object corresponds to the Boolean twin of the derivative of a function of real variables: it studies the variation along a given direction. The derivatives of a Boolean function $f\colon \mathbb{F}_2^n \to \mathbb{F}_2$ also have strong interactions with the derivatives of its ANF $P \in \mathbb{F}_2[X_0, \ldots, X_{n-1}]$, that is, derivatives of *polynomials*.

For instance, the derivative of f with respect to $(1,0,\dots 0)$ has as ANF, the derivative of the ANF of f with respect to x_0: $D_{(1,0,\dots,0)}f = \frac{\partial P}{\partial X_0}$. However the two objects should not get mixed up. For instance, contrary to derivatives of polynomials, derivatives of vectorial Boolean functions do not satisfy the classical chain rule formula [CCP22]. ▷

The set of solutions of Eq. (2.9) is the preimage of $\{\Delta^{\text{out}}\}$ by $D_{\Delta^{\text{in}}}F$. We denote it by $Z_F^{\text{diff}}(\Delta^{\text{in}}, \Delta^{\text{out}})$ and its cardinality by $\delta_F(\Delta^{\text{in}}, \Delta^{\text{out}})$:

$$Z_F^{\text{diff}}(\Delta^{\text{in}}, \Delta^{\text{out}}) := (D_{\Delta^{\text{in}}}F)^{-1}\left(\left\{\Delta^{\text{out}}\right\}\right) = \left\{x \in \mathbb{F}_2^n, F(x+\Delta^{\text{in}}) + F(x) = \Delta^{\text{out}}\right\},$$

$$\delta_F(\Delta^{\text{in}}, \Delta^{\text{out}}) := \left|Z_F^{\text{diff}}(\Delta^{\text{in}}, \Delta^{\text{out}})\right|.$$

The values $\Delta^{\text{in}}, \Delta^{\text{out}}$ for which $\delta_F(\Delta^{\text{in}}, \Delta^{\text{out}})$ is maximal are therefore the biggest threats in terms of differential attacks. This maximal value serves as a security parameter.

Definition 2.25 (Differential uniformity [Nyb94])**.** Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$. The *differential uniformity* is the value denoted by δ_F that is defined by:

$$\delta_F := \max_{\Delta^{\text{in}} \in \mathbb{F}_2^{n*}, \Delta^{\text{out}} \in \mathbb{F}_2^m} \delta_F(\Delta^{\text{in}}, \Delta^{\text{out}}).$$

▷

As shown later in Definition 2.37, the table containing the values $\delta_F(\Delta^{\text{in}}, \Delta^{\text{out}})$ for all $\Delta^{\text{in}}, \Delta^{\text{out}}$ is also relevant when it comes to the precise analysis of the different components of a block cipher.

2.3.3.c Optimal resistance to differential attacks

Ideally, we would like δ_F to be as low as possible. For any Δ^{in}, as $D_{\Delta^{\text{in}}}F$ is well defined, there must exist Δ^{out} such that $\delta_F(\Delta^{\text{in}}, \Delta^{\text{out}}) > 0$. Therefore $\delta_F > 0$. Furthermore, let x be a solution to Eq. (2.9) for a non-zero Δ^{in}. Then $x + \Delta^{\text{in}} \neq x$ (as $\Delta^{\text{in}} \neq 0$), and $x + \Delta^{\text{in}}$ is also a solution of Eq. (2.9). Indeed,

$$F\left((x+\Delta^{\text{in}}) + \Delta^{\text{in}}\right) + F(x+\Delta^{\text{in}}) = F(x) + F(x+\Delta^{\text{in}}) = \Delta^{\text{out}}. \tag{2.11}$$

This proves that the solutions of Eq. (2.9) always come by pair, and as such, any $\delta_F(\Delta^{\text{in}}, \Delta^{\text{out}})$ is an even number, and so is δ_F. This implies that the optimal resistance to differential attacks is obtained when $\delta_F = 2$. We define those optimal functions as follows.

Definition 2.26 (Almost Perfect Non-linear function [NK93])**.** Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$. F is called *almost perfect non-linear* (APN) if it satisfies $\delta_F = 2$. ▷

APN functions are one of the main topic of Chapter 6. For now let us just clarify the underlying notion of linearity that is mentioned in the naming thanks to the following lemma.

Lemma 2.27 (Characterization of affine functions in terms of $\delta_F(\Delta^{\text{in}}, \Delta^{\text{out}})$). *Let $F \colon \mathbb{F}_2^n \to \mathbb{F}_2^m$. Then F is an affine function if and only if it satisfies:*

$$\forall\ \Delta^{\text{in}} \in \mathbb{F}_2^n, \forall\ \Delta^{\text{out}} \in \mathbb{F}_2^m, \quad \delta_F(\Delta^{\text{in}}, \Delta^{\text{out}}) = \begin{cases} 2^n & \textit{if } \Delta^{\text{out}} = F(\Delta^{\text{in}}) + F(0), \\ 0 & \textit{otherwise.} \end{cases}$$

Proof. A function F is affine if and only if:

$$\forall\ x, y \in \mathbb{F}_2^n, \quad F(x+y) + F(x) + F(y) + F(0) = 0.$$

By using Δ^{in} instead of x, this is equivalent to:

$$\forall\ \Delta^{\text{in}} \in \mathbb{F}_2^n, \quad D_{\Delta^{\text{in}}} F = F(\Delta^{\text{in}}) + F(0).$$

In other words, a function is affine if and only if all its derivatives are constant. Equivalently, F is affine if and only if given any $\Delta^{\text{in}} \in \mathbb{F}_2^n$, $\Delta^{\text{out}} \in \mathbb{F}_2^m$, the equation $D_{\Delta^{\text{in}}} F(x) = \Delta^{\text{out}}$ holds either for all x or has no solution. □

In the light of Lemma 2.27, we observe that an APN function behaves in a strictly opposite way compared to a linear (or affine) one. For a given Δ^{in}, if F is linear, the image of $D_{\Delta^{\text{in}}} F$ contains a single value Δ^{out}, which therefore has 2^n preimages. On the contrary, if F is APN, the non-empty preimages of $D_{\Delta^{\text{in}}}(F)$ must be of size exactly 2, so $D_{\Delta^{\text{in}}} F(\mathbb{F}_2^n)$ is of cardinality exactly 2^{n-1}. Finally, the adverb "almost" indicates that δ_F cannot be equal to 1. Indeed, if we extend the definition of differential uniformity to functions $F \colon G \to H$ between two Abelian groups G, H, then for any $\Delta^{\text{in}} \in G$, it holds that:

$$\sum_{\Delta^{\text{out}} \in H} \delta_F(\Delta^{\text{in}}, \Delta^{\text{out}}) = |G|\,.$$

This implies that for any $\Delta^{\text{in}} \in G, \Delta^{\text{out}} \in H$, $\delta_F(\Delta^{\text{in}}, \Delta^{\text{out}}) \geq \frac{G}{H}$ and therefore $\delta_F \geq \frac{G}{H}$. For this reason, a function is said to be *perfect nonlinear* [Nyb91] if it satisfies $\delta_F = \frac{|G|}{|H|}$. However, for Boolean functions from $\mathbb{F}_2^n$ to $\mathbb{F}_2^m$, perfect nonlinear functions can only exist if $2m \leq n$ [Nyb91].

2.3.3.d Finding differential distinguisher

Unkeyed function. In practice, for a cryptographic function $F \colon \mathbb{F}_2^n \to \mathbb{F}_2^m$, it is impossible to compute $\delta_F(\Delta^{\text{in}}, \Delta^{\text{out}})$ for any $\Delta^{\text{in}} \in \mathbb{F}_2^n$ and $\Delta^{\text{out}} \in \mathbb{F}_2^m$ due to the large size of n. However, we can leverage the iterated constructions, that most of the symmetric primitives follow, to approximate the number of solutions.

To simplify the presentation, let us consider that the domain and codomain of each involved function are $\mathbb{F}_2^n$. More precisely, let $F \colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ be a function that can be decomposed as $F = F^{(R-1)} \circ \cdots \circ F^{(0)}$ where $F^{(r)} \colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ for any $r \in [\![0, R-1]\!]$. Let $\Delta^{(0)}, \Delta^{(R)} \in \mathbb{F}_2^n$ be a pair of input/output differences. For any $x \in \mathbb{F}_2^n$, let us denote the *intermediate values* by $x^{(0)} := x$, $y^{(0)} := x + \Delta^{(0)}$, and for any $r \geq 0$, $x^{(r+1)} := F^{(r)}(x^{(r)})$ and $y^{(r+1)} := F^{(r)}(y^{(r)})$.

Remark 2.28. Even if it does not appear in the notation, we insist on the fact that for any r, the value $y^{(r)}$ *depends* on both x and $\Delta^{(0)}$. ▷

With this notation, and by using the value of the *intermediate differences*, we can partition the set of solutions of Eq. (2.9) as:

$$\begin{aligned}
Z_F^{\text{diff}}(\Delta^{\text{in}}, \Delta^{\text{out}}) &= \left\{x \in \mathbb{F}_2^n,\ F(x) + F(x + \Delta^{(0)}) = \Delta^{(R)}\right\} \\
&= \left\{x \in \mathbb{F}_2^n,\ x^{(R)} + y^{(R)} = \Delta^{(R)}\right\} \\
&= \bigsqcup_{\Delta^{(R-1)} \in \mathbb{F}_2^n} \left\{x \in \mathbb{F}_2^n,\ x^{(R)} + y^{(R)} = \Delta^{(R)}, x^{(R-1)} + y^{(R-1)} = \Delta^{(R-1)}\right\} \\
&\ \vdots \\
&= \bigsqcup_{\Delta^{(1)},\dots,\Delta^{(R-1)} \in \mathbb{F}_2^n} \left\{x \in \mathbb{F}_2^n, \forall r \in [\![1, R]\!], x^{(r)} + y^{(r)} = \Delta^{(r)}\right\}.
\end{aligned}$$

The number of solutions can therefore be computed as:

$$\delta_F(\Delta^{(0)}, \Delta^{(R)}) = \sum_{\Delta^{(1)},\dots,\Delta^{(R-1)} \in \mathbb{F}_2^n} \left|\left\{x \in \mathbb{F}_2^n,\ \forall\, r \in [\![1, R]\!]\ \ x^{(r)} + y^{(r)} = \Delta^{(r)}\right\}\right|. \tag{2.12}$$

In this context, we traditionally use the following vocabulary.

Definition 2.29 (Differential, differential trail, fixed-key differential probability)**.** A pair of differences $(\Delta^{(0)}, \Delta^{(R)}) \in (\mathbb{F}_2^n)^2$ is called a *differential* and is denoted by $\Delta^{(0)} \to \Delta^{(R)}$. Similarly a vector of differences $(\Delta^{(0)}, \Delta^{(1)}, \dots, \Delta^{(R)}) \in (\mathbb{F}_2^n)^{R+1}$ is called a *differential trail* (or *differential characteristic*) and is denoted by $\Delta^{(0)} \to \Delta^{(1)} \to \dots \to \Delta^{(R)}$.

By *differential probability* of the differential $\Delta^{(0)} \to \Delta^{(R)}$ over F, we mean the value $\mathbb{P}\left[\Delta^{(0)} \xrightarrow{F} \Delta^{(R)}\right]$ that is defined by:

$$\mathbb{P}\left[\Delta^{(0)} \xrightarrow{F} \Delta^{(R)}\right] := \frac{\delta_F(\Delta^{(0)}, \Delta^{(R)})}{2^n}.$$

In the same way, the *differential probability of the trail* $\Delta^{(0)} \to \Delta^{(1)} \to \dots \to \Delta^{(R)}$ over F (with respect to the decomposition $F = F^{(R-1)} \circ \dots \circ F^{(0)}$) is defined by:

$$\mathbb{P}\left[\Delta^{(0)} \xrightarrow{F^{(0)}} \Delta^{(1)} \to \dots \xrightarrow{F^{(R-1)}} \Delta^{(R)}\right] := \frac{\left|\left\{x \in \mathbb{F}_2^n,\ \forall\, r \in [\![1, R]\!]\ \ x^{(r)} + y^{(r)} = \Delta^{(r)}\right\}\right|}{2^n}$$

▷

With this new notation in mind, Eq. (2.12) can be formulated as:

$$\mathbb{P}\left[\Delta^{(0)} \xrightarrow{F} \Delta^{(R)}\right] = \sum_{\Delta^{(1)},\dots,\Delta^{(R-1)} \in \mathbb{F}_2^n} \mathbb{P}\left[\Delta^{(0)} \xrightarrow{F^{(0)}} \Delta^{(1)} \to \dots \xrightarrow{F^{(R-1)}} \Delta^{(R)}\right]. \tag{2.13}$$

As each of the term in this sum is positive, the differential probability of $\Delta^{(0)} \to \Delta^{(R)}$ can be lower bounded by the sum over *any* number of differential probabilities of trails $\Delta^{(0)} \to \Delta^{(1)} \to \cdots \to \Delta^{(R)}$. Indeed, for any $\mathcal{D} \subset (\mathbb{F}_2^n)^{R-1}$ it holds that:

$$\mathbb{P}\left[\Delta^{(0)} \xrightarrow{F} \Delta^{(R)}\right] \geq \sum_{(\Delta^{(1)},\ldots,\Delta^{(R-1)}) \in \mathcal{D}} \mathbb{P}\left[\Delta^{(0)} \xrightarrow{F^{(0)}} \Delta^{(1)} \to \cdots \xrightarrow{F^{(R-1)}} \Delta^{(R)}\right].$$

Block ciphers. When considering a block cipher $\mathcal{E} = (E_k \colon \mathbb{F}_2^n \xrightarrow{\sim} \mathbb{F}_2^n)_{k \in \mathbb{F}_2^\kappa}$, we are instead interested in understanding the differential probability of each E_k. We therefore would like to understand the distribution of the sequence $\left(\mathbb{P}\left[\Delta^{(0)} \xrightarrow{E_k} \Delta^{(R)}\right]\right)_{k \in \mathbb{F}_2^\kappa}$. In the usual case where $\kappa = 128$, this gigantic sequence might be hard to grasp. As often, the average value might help better understanding it.

Definition 2.30 (Expected differential probability [LMM91])**.** Let $\mathcal{E}$ be a block cipher: $\mathcal{E} = (E_k \colon \mathbb{F}_2^n \xrightarrow{\sim} \mathbb{F}_2^n)_{k \in \mathbb{F}_2^\kappa}$. Let $R \geq 1$. Let us suppose that, for any $k \in \mathbb{F}_2^\kappa$, E_k can be decomposed as $E_k = F_k^{(R-1)} \circ \cdots \circ F_k^{(0)}$ and denote by $\mathcal{F}^{(r)}$ the family of the r-th round functions: $\mathcal{F}^{(r)} := (F_k^{(r)})_{k \in \mathbb{F}_2^\kappa}$. Let $\Delta^{(0)}, \ldots, \Delta^{(R)} \in \mathbb{F}_2^n$. The *expected differential probability* of the differential $\Delta^{(0)} \to \Delta^{(R)}$ over $\mathcal{E}$ is the averaged differential probability of $\Delta^{(0)} \to \Delta^{(R)}$ over all keys. It is denoted by $\mathbb{E}\left[\Delta^{(0)} \xrightarrow{\mathcal{E}} \Delta^{(R)}\right]$ and defined by:

$$\mathbb{E}\left[\Delta^{(0)} \xrightarrow{\mathcal{E}} \Delta^{(R)}\right] := \frac{1}{2^\kappa} \sum_{k \in \mathbb{F}_2^\kappa} \mathbb{P}\left[\Delta^{(0)} \xrightarrow{E_k} \Delta^{(R)}\right].$$

Similarly, the expected differential probability of the trail $\Delta^{(0)} \to \Delta^{(1)} \to \cdots \to \Delta^{(R)}$ is defined by:

$$\mathbb{E}\left[\Delta^{(0)} \xrightarrow{\mathcal{F}^{(0)}} \Delta^{(1)} \to \cdots \xrightarrow{\mathcal{F}^{(R-1)}} \Delta^{(R)}\right] := \frac{1}{2^\kappa} \sum_{k \in \mathbb{F}_2^\kappa} \mathbb{P}\left[\Delta^{(0)} \xrightarrow{F_k^{(0)}} \Delta^{(1)} \to \cdots \xrightarrow{F_k^{(R-1)}} \Delta^{(R)}\right].$$

▷

Again, the expected differential probability of $\Delta^{(0)} \to \Delta^{(R)}$ can be expressed as the sum of expected differential probabilities of the associated trails, and lower bounded by partial sums:

$$\mathbb{E}\left[\Delta^{(0)} \xrightarrow{\mathcal{E}} \Delta^{(R)}\right] = \sum_{\Delta^{(1)},\ldots,\Delta^{(R-1)} \in \mathbb{F}_2^n} \mathbb{E}\left[\Delta^{(0)} \xrightarrow{\mathcal{F}^{(0)}} \Delta^{(1)} \to \cdots \xrightarrow{\mathcal{F}^{(R-1)}} \Delta^{(R)}\right], \text{ and}$$

$$\mathbb{E}\left[\Delta^{(0)} \xrightarrow{\mathcal{E}} \Delta^{(R)}\right] \geq \sum_{\Delta^{(1)},\ldots,\Delta^{(R-1)} \in \mathcal{D}} \mathbb{E}\left[\Delta^{(0)} \xrightarrow{\mathcal{F}^{(0)}} \Delta^{(1)} \to \cdots \xrightarrow{\mathcal{F}^{(R-1)}} \Delta^{(R)}\right]. \tag{2.14}$$

The advantage with the expected differential probability of a trail is that it can be easily computed in the case of a key-alternating block cipher *with independent round keys*.

Proposition 2.31 (Expected differential probability for key-alternating block cipher [LMM91])**.** *Let $R \geq 1$, $F^{(0)}, \dots, F^{(R-1)} \colon \mathbb{F}_2^n \to \mathbb{F}_2^n$, and let $\mathcal{E}$ be a block cipher such that $\mathcal{E} = (E_k \colon \mathbb{F}_2^n \xrightarrow{\sim} \mathbb{F}_2^n)_{k \in (\mathbb{F}_2^n)^{R-1}}$ and such that for any $k = \left(k^{(1)}, \dots, k^{(R-1)}\right) \in (\mathbb{F}_2^n)^{R-1}$, E_k can be decomposed as follows:*

$$E_k = F^{(R-1)} \circ T_{k^{(R-1)}} \circ \cdots \circ F^{(1)} \circ T_{k^{(1)}} \circ F^{(0)}.$$

Let $\mathcal{F}^{(0)} := (F^{(0)})_{k \in (\mathbb{F}_2^n)^{R-1}}$, and let $\mathcal{F}^{(r)} := (F^{(r)} \circ T_{k^{(r)}})_{k \in (\mathbb{F}_2^n)^{R-1}}$ for any $r \geq 1$. Then, for any trail $\Delta^{(0)} \to \Delta^{(1)} \to \cdots \to \Delta^{(R)}$, it holds that:

$$\mathbb{E}\left[\Delta^{(0)} \xrightarrow{\mathcal{F}^{(0)}} \Delta^{(1)} \to \cdots \xrightarrow{\mathcal{F}^{(R-1)}} \Delta^{(R)}\right] = \prod_{r=0}^{R-1} \mathbb{P}\left[\Delta^{(r)} \xrightarrow{F^{(r)}} \Delta^{(r+1)}\right].$$

Remark 2.32. Before going further, we observe that for any $x, c, \Delta^{\text{in}}, \Delta^{\text{out}} \in \mathbb{F}_2^n$, and any function $F \colon \mathbb{F}_2^n \to \mathbb{F}_2^n$, x satisfies $T_c \circ F(x + \Delta^{\text{in}}) + T_c \circ F(x) = \Delta^{\text{out}}$ if and only if x satisfies:

$$F(x + \Delta^{\text{in}}) + c + F(x) + c = \Delta^{\text{out}} \iff F(x + \Delta^{\text{in}}) + F(x) = \Delta^{\text{out}}.$$

In other words, an outer constant (or key) addition never impacts a differential property. This is the reason why the cipher considered in Proposition 2.31 does not finish with a key addition. Stated otherwise, the scope of Proposition 2.31 can be extended to ciphers with an outer key addition. ▷

Proof of Proposition 2.31. Let us prove it by induction on the number of rounds R.

The case $R = 1$. If $R = 1$, then the considered "cipher" is a family made of $(2^n)^{R-1} = 1$ bijection, which is F_0. A differential trail of length 1 is nothing more than a differential and the associated expected differential degenerates into:

$$\mathbb{E}\left[\Delta^{(0)} \xrightarrow{\mathcal{F}^{(0)}} \Delta^{(1)}\right] = \mathbb{P}\left[\Delta^{(0)} \xrightarrow{F_0} \Delta^{(1)}\right],$$

as the average over a sequence of size 1 is nothing more than the single value itself.

The general case. For any $R \geq 1$, let us define the set Y_R as the set containing all vectors $\left(x, k^{(1)}, \cdots, k^{(R-1)}\right) \in (\mathbb{F}_2^n)^R$ that satisfy the following system of equations:

$$\begin{cases} F^{(0)}\left(x^{(0)}\right) + F^{(0)}\left(x^{(0)} + \Delta^{(0)}\right) & = \Delta^{(1)} \\ F^{(1)}\left(x^{(1)} + \Delta^{(1)} + k^{(1)}\right) + F^{(1)}\left(x^{(1)} + k^{(1)}\right) & = \Delta^{(2)} \\ \vdots & \\ F^{(R-1)}\left(x^{(R-1)} + \Delta^{(R-1)} + k^{(R-1)}\right) + F^{(R-1)}\left(x^{(R-1)} + k^{(R-1)}\right) & = \Delta^{(R)}, \end{cases}$$

where $x^{(0)} := x$ and $x^{(r+1)} := F^{(r)}\left(x^{(r)}\right)$ for any r. By construction, the cardinality of Y_R satisfies:

$$2^{nR} \cdot \mathbb{E}\left[\Delta^{(0)} \xrightarrow{\mathcal{F}^{(0)}} \Delta^{(1)} \to \cdots \xrightarrow{\mathcal{F}^{(R-1)}} \Delta^{(R)}\right] = |Y_R|.$$

It is therefore equivalent to prove that:

$$|Y_R| = \prod_{r=0}^{R-1} \left| Z^{\text{diff}}_{F^{(r)}} \left(\Delta^{(r)}, \Delta^{(r+1)} \right) \right|. \tag{2.15}$$

Let $R \geq 1$ and let us suppose that Eq. (2.15) is satisfied by R. We observe that $|Y_{R+1}|$ can be expressed as:

$$|Y_{R+1}| = \sum_{y \in Y_R} |Z_y| ,$$

where for any $y = \left(x, k^{(0)}, \dots, k^{(R-1)}\right)$, the set Z_y is defined by:

$$Z_y = \left\{ k^{(R)} \in \mathbb{F}_2^n, F^{(R)} \left(x^{(R)} + \Delta^{(R)} + k^{(R)} \right) + F^{(R)} \left(x^{(R)} + k^{(R)} \right) = \Delta^{(R+1)} \right\}.$$

Note that while y does not explicitly appear in the definition of Z_y, the value $x^{(R)}$ does appear and does depend on y. Let $y = \left(x, k^{(0)}, \dots, k^{(R-1)}\right)$ be fixed, so that $x^{(R)}$ is also fixed. By using $z \leftarrow x^{(R)} + k^{(R)}$ as a change of variables, we obtain:

$$Z_y = \left\{ z \in \mathbb{F}_2^n, F^{(R)} \left(z + \Delta^{(R)} \right) + F^{(R)} (z) = \Delta^{(R+1)} \right\} = Z^{\text{diff}}_{F^{(R)}} \left(\Delta^{(R)}, \Delta^{(R+1)} \right),$$

and Z_y is therefore independent of y. We finally obtain:

$$\begin{aligned}
|Y_{R+1}| &= \sum_{y \in Y_R} |Z_y| \\
&= \left| Z^{\text{diff}}_{F^{(R)}} \left(\Delta^{(R)}, \Delta^{(R+1)} \right) \right| \sum_{y \in Y_R} 1 \\
&= \left| Z^{\text{diff}}_{F^{(R)}} \left(\Delta^{(R)}, \Delta^{(R+1)} \right) \right| \times |Y_R| \\
&= Z^{\text{diff}}_{F^{(R)}} \left(\Delta^{(R)}, \Delta^{(R+1)} \right) \left(\prod_{r=0}^{R-1} \left| Z^{\text{diff}}_{F^{(r)}} \left(\Delta^{(r)}, \Delta^{(r+1)} \right) \right| \right) \\
&= \prod_{r=0}^{R} \left| Z^{\text{diff}}_{F^{(r)}} \left(\Delta^{(r)}, \Delta^{(r+1)} \right) \right| ,
\end{aligned}$$

where we use the induction hypothesis to obtain the fourth equality. □

Thanks to Proposition 2.31, the computation of the expected differential probability of a trail comes back to computing the probability of some unkeyed subfunctions. Contrary to the intricate general case that was presented above, the unkeyed round functions used in a block cipher are usually *very simple* and their differential probabilities can easily be computed.

Proposition 2.33 (Differential probabilities of round functions)**.** *Let $n = m \times s$. Let $S^{(0)}, \dots, S^{(s-1)} \colon \mathbb{F}_2^m \to \mathbb{F}_2^m$. Let $\mathcal{S} \colon (\mathbb{F}_2^m)^s \to (\mathbb{F}_2^m)^s$ be the function defined by*

$$\mathcal{S} \colon (x_0, \dots, x_{s-1}) \to (S^{(0)}(x_0), \dots S^{(s-1)}(x_{s-1})).$$

Let $L\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ be linear. Let $\Delta^{\text{in}}, \Delta^{\text{out}} \in \mathbb{F}_2^n$ be such that $\Delta^{\text{in}} =: (\Delta_0^{\text{in}}, \dots, \Delta_{s-1}^{\text{in}})$ and $\Delta^{\text{out}} =: (\Delta_0^{\text{out}}, \dots, \Delta_{s-1}^{\text{out}})$ where $\Delta_i^{\text{in}}, \Delta_i^{\text{out}} \in \mathbb{F}_2^m$ for any i. Then:

$$\mathbb{P}\left[\Delta^{\text{in}} \xrightarrow{\mathcal{S}} \Delta^{\text{out}}\right] = \prod_{i=0}^{s-1} \mathbb{P}\left[\Delta_i^{\text{in}} \xrightarrow{S^{(i)}} \Delta_i^{\text{out}}\right],$$

$$\textit{and} \quad \mathbb{P}\left[\Delta^{\text{in}} \xrightarrow{L} \Delta^{\text{out}}\right] = \begin{cases} 1 & \textit{if } \Delta^{\text{out}} = L(\Delta^{\text{in}}) \\ 0 & \textit{otherwise.} \end{cases}$$

Proof. The formula for L was already proven in Lemma 2.27. For the Sbox layer $\mathcal{S}$, we observe that for any $x = (x_0, \dots, x_{s-1}) \in \mathbb{F}_2^n$, we have

$$\mathcal{S}(x)+\mathcal{S}(x+\Delta^{\text{in}}) = \Delta^{\text{out}} \iff \forall\, i \in [\![0, s-1]\!]\,,\ S^{(i)}(x_i)+S^{(i)}(x_i+\Delta_i^{\text{in}}) = \Delta_i^{\text{out}}.$$

We immediately deduce that $\delta_{\mathcal{S}}(\Delta^{\text{in}}, \Delta^{\text{out}}) = \prod_{i=0}^{s-1} \delta_{S^{(i)}}(\Delta_i^{\text{in}}, \Delta_i^{\text{out}})$, which implies the announced equality. □

2.3.3.e Main assumptions

We briefly sum up the common method used to assess the security of a key-alternating block cipher or a cryptographic primitive with respect to differential cryptanalysis. This is done by emphasizing the assumptions made at each step.

Let $\mathcal{E} = (E_k\colon \mathbb{F}_2^n \to \mathbb{F}_2^n)_{k\in\mathbb{F}_2^\kappa}$ be an r-round key-alternating block cipher and KS: $\mathbb{F}_2^\kappa \to (\mathbb{F}_2^n)^r$ be its key schedule algorithm. The expected differential probability of a trail over $\mathcal{E}$ can be *approximated* (using Proposition 2.31) by making the following assumption.

Assumption 2.34. *The expected differential probability of $\Delta^{(0)} \to \Delta^{(1)} \to \dots \to \Delta^{(R)}$ over $\mathcal{E}$ is close to the expected differential probability of $\Delta^{(0)} \to \Delta^{(1)} \to \dots \to \Delta^{(R)}$ over $\widetilde{\mathcal{E}} = (\widetilde{E}_k\colon \mathbb{F}_2^n \to \mathbb{F}_2^n)_{k\in(\mathbb{F}_2^n)^R}$ where $\widetilde{\mathcal{E}}$ is derived from $\mathcal{E}$ by ignoring* KS *and allowing each round key to take all possible values.*

From there, an attacker searches for one or some trails with the same input and output differences $\Delta^{(0)}, \Delta^{(R)}$, and for which $\mathbb{E}\left[\Delta^{(0)} \xrightarrow{F^{(0)}} \Delta^{(1)} \to \cdots \xrightarrow{F^{(R-1)}} \Delta^{(R)}\right]$ is high. However, with a blackbox access to the keyed cipher, no information from the intermediate differences can be leveraged. This is the reason why Eq. (2.14) is usually used to estimate $\mathbb{E}\left[\Delta^{(0)} \xrightarrow{\mathcal{E}} \Delta^{(R)}\right]$. The more trails are considered, the tighter is the lower bound. However for $\mathbb{E}\left[\Delta^{(0)} \xrightarrow{\mathcal{E}} \Delta^{(R)}\right]$ to be representative of the sequence $\left(\mathbb{P}\left[\Delta^{(0)} \xrightarrow{E_k} \Delta^{(R)}\right]\right)_{k\in\mathbb{F}_2^\kappa}$, the so-called *stochastic equivalence hypothesis* must be assumed.

Assumption 2.35 (Stochastic equivalence hypothesis [LMM91])**.** *For any (or most) $k \in \mathbb{F}_2^\kappa$, $\mathbb{P}\left[\Delta^{(0)} \xrightarrow{E_k} \Delta^{(R)}\right]$ is close to the expected differential probability $\mathbb{E}\left[\Delta^{(0)} \xrightarrow{\mathcal{E}} \Delta^{(R)}\right]$.*

Both assumptions are impactful and should be used with caution. They were presented by Lai, Massey & Murphy [LMM91] more than 30 years ago, together with a slightly more general[2] statement than Proposition 2.31. Subsequently, examples where it greatly fails were quickly exhibited [Knu93]. This inaccurate methodology is however still at use, as new tools to cope with the study of 2^κ functions at once are lacking. Examples of either, mean deviation or, lack of independence still abound today [Can+17, AK19, PT22, BR22].

Hash functions. A probably more astonishing fact is that the same methodology is used to approximate the differential probability of a differential over a *single* unkeyed cryptographic function such as a hash function. Non-exhaustive examples are for instance the cryptanalysis of MD4 [Dob98], or the more recent one of Keccak [DDS14]. Indeed, by assuming that *fictive*[3] independent round keys can be added between each round of the hash function, the average behaviour over all keys can be studied. In that case, the hash function corresponds to the function where all round keys are zero. Then again, under Assumption 2.35, its actual differential probability is likely to match with the average behaviour.

Fixed-key cryptanalysis. The interest for analyzing primitives in the fixed-key model has recently been freshened by the work of Beyne & Rijmen [BR22] where the authors study the function $F \times F := (x, y) \mapsto (F(x), F(y))$. This function encapsulates all information about differences, but above all, it keeps track of the values of each input/output pair. By *partially* applying a Fourier transform on the transition matrix (see Section 2.2.2.b) of $F \times F$, the authors obtain the *quasidifferential matrix* $\mathbf{D}^F$, which, as the LAT, inherits of the already-mentioned composition property. By looking at the coefficients of the product of matrices $\mathbf{D}^{F^{(r-1)}} \cdots \mathbf{D}^{F^{(0)}}$, the authors obtain a closed formula for the differential probability $\mathbb{P}\left[\Delta^{(0)} \xrightarrow{F^{(0)}} \Delta^{(1)} \to \cdots \xrightarrow{F^{(r-1)}} \Delta^{(R)}\right]$. This exact formula is based on more intricate trails. Because $\mathbf{D}^F \in \mathbf{M}_{2^{2n}}(\mathbb{F}_2)$ (compared to the LAT which lies in $\mathbf{M}_{2^n}(\mathbb{F}_2)$), the practical applicability of this technique on a large scale remains an open problem.

[2] The formula given in Proposition 2.31 can be extended to the case of *Markov ciphers*, which, in practice, often comes back to the situation described in Proposition 2.31.

[3] As already noted in Remark 1.6, a cryptographic hash function is a *public* function which therefore does not depend on any key.

The designer point-of-view. Finally, a designer who wishes to build a strong primitive faces (almost) the same challenges as an attacker. In particular, the famous wide-trail strategy [DR01] leverages the differential uniformity of the Sbox and the branch number of the linear layer to bound from above $\mathbb{E}\left[\Delta^{(0)} \xrightarrow{\mathcal{F}^{(0)}} \Delta^{(1)} \to \cdots \xrightarrow{\mathcal{F}^{(R-1)}} \Delta^{(R)}\right]$ for any trail $\Delta^{(0)} \to \Delta^{(1)} \to \cdots \to \Delta^{(R)}$. In the particular case where the linear layer has a maximal branch number, it also provides an upper bound for the expected differential probability $\mathbb{E}\left[\Delta^{\text{in}} \xrightarrow{\mathcal{F}^{(1)} \circ \mathcal{F}^{(0)}} \Delta^{\text{out}}\right]$ of any *differential* over two rounds constructions with a key addition in the middle [DR02, Section B.2]. However, in more general contexts, it is hard to provide such guarantees. In particular, it can happen (for some keys) that many trails with a small yet non-zero probability sum up in a differential which reaches a high probability. This is called a *clustering effect*. As a side effect of the work presented in Chapter 5, such a clustering phenomenon is exposed for modified versions of Midori. Another recent and good example is the work of Leurent, Pernot & Schrottenloher [LPS21]. This points out another challenged assumption that is due to the lack of available security arguments, and that is commonly made when designing a primitive.

Assumption 2.36. *Bounding from above the expected differential probability for all trails $\Delta^{(0)} \to \Delta^{(1)} \to \cdots \to \Delta^{(R)}$ is sufficient to guarantee a low differential probability for any differential $\Delta^{(0)} \to \Delta^{(R)}$.*

2.3.3.f The differential distribution table

As highlighted by Proposition 2.33, the only non-deterministic differential transition that occurs in a cipher comes from the Sbox layer. As already hinted, by choosing an Sbox S such that δ_S is low-enough, we can ensure that the value $\mathbb{P}\left[\Delta^{(0)} \xrightarrow{\mathcal{S}} \Delta^{(R)}\right]$ for any differential $\Delta^{(0)} \to \Delta^{(R)}$ is not too high. The computation of δ_S is possible in that case due to the small domain and codomain of S. However, because of Proposition 2.31, the value $\mathbb{E}\left[\Delta^{(0)} \xrightarrow{\mathcal{F}^{(0)}} \Delta^{(1)} \to \cdots \xrightarrow{\mathcal{F}^{(r-1)}} \Delta^{(R)}\right]$ can reach a not-so-low value, because the probabilities $\mathbb{P}\left[\Delta^{(r)} \xrightarrow{F^{(t)}} \Delta^{(r+1)}\right]$ that are multiplied are all medium values. This is the reason why not only δ_F matters, but also the specific values $\Delta^{\text{in}}, \Delta^{\text{out}}$ for which $\delta_F(\Delta^{\text{in}}, \Delta^{\text{out}})$ is quite high, and more generally, all values $\delta_F(\Delta^{\text{in}}, \Delta^{\text{out}})$. Those values are stored in the difference distribution table of S.

Definition 2.37 (Difference Distribution Table[4] (DDT) [BS91b])**.** Let $F \colon \mathbb{F}_2^n \to \mathbb{F}_2^m$. The *difference distribution table* (DDT) of F is the $2^n \times 2^m$ integer matrix that stores the number of solutions of each differential equation:

$$\mathrm{DDT}_F := \left(\delta_F(\Delta^{\text{in}}, \Delta^{\text{out}})\right)_{\Delta^{\text{in}} \in \mathbb{F}_2^n, \Delta^{\text{out}} \in \mathbb{F}_2^m}.$$

[4] The DDT is originally called *XOR distribution table* in the seminal work of Biham & Shamir [BS91b].

▷

While the definition is generic, this object can only be manipulated for small values of n and m.

Example 2.38 (DDT of the Sbox of Ascon)**.** As for the LAT, the DDT of the Sbox of Ascon can be graphically represented using a grayscale, as shown in Figure 2.3. In particular, we observe that for any $\Delta^{\text{in}}, \Delta^{\text{out}} \in \mathbb{F}_2^5$, $\delta_S(\Delta^{\text{in}}, \Delta^{\text{out}}) \in \{0, 2, 4, 8\}$, and that $\delta_S = 8$.

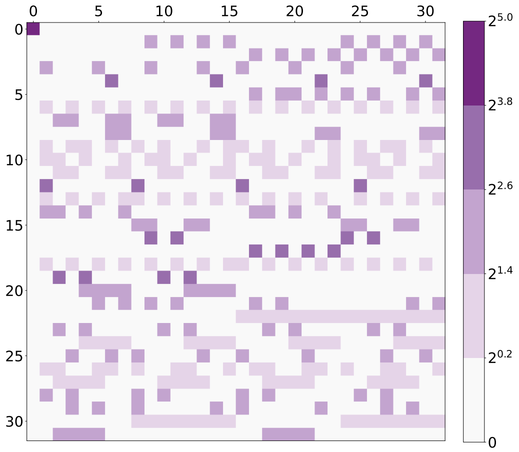


Figure 2.3: DDT of the Sbox of Ascon.

▷

It is therefore considered a good practice to design an Sbox whose DDT has as few coefficients equal to its differential uniformity as possibles. Many open problems related to the existence or design of good Sbox exist. As an example, we do not know if there exists an APN bijective Sbox $S\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ for an even $n > 6$. This problem is further discussed in Chapter 6.

2.3.4 Linearity

2.3.4.a Cryptographic context

Linear cryptanalysis is one of the two most important kinds of statistical attacks against block ciphers. This technique is credited to Matsui [Mat94], but its infancy dates back to 1991 with a work of Tardy-Corfdir & Gilbert [TG92]. It consists in finding highly-probable affine relations between the input and the output of a cryptographic function $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$, that is, equations of the following form with many solutions $x \in \mathbb{F}_2^n$:

$$\alpha \cdot x = \beta \cdot F(x) + \varepsilon \quad \iff \quad \sum_{i=0}^{n-1} \alpha_i x_i = \sum_{i=0}^{m-1} \beta_i F_i(x) + \varepsilon, \tag{2.16}$$

where $\alpha \in \mathbb{F}_2^n, \beta \in \mathbb{F}_2^m$, $\varepsilon \in \mathbb{F}_2$. Such relations enable us to distinguish F from a random function. Indeed, if F is drawn uniformly at random among all functions, then for any $\alpha, \beta, \varepsilon$, Eq. (2.16) is expected to hold for half of the values $x \in \mathbb{F}_2^n$, that is, with probability $\frac{1}{2}$. Note that it is sufficient to consider $\varepsilon = 0$. In that case, we are instead interested in linear equations in input and output bits that hold either for a lot or very few $x \in \mathbb{F}_2^n$. The more the number of solutions deviates from $\frac{2^n}{2}$, the easier it is for an attacker to distinguish F from a random function.

For a block cipher $\mathcal{E} = (E_k \colon \mathbb{F}_2^n \xrightarrow{\sim} \mathbb{F}_2^n)_{k \in \mathbb{F}_2^\kappa}$, as in the differential case, such a distinguisher is built if there exist $\alpha, \beta \in \mathbb{F}_2^n$, $\varepsilon \in \mathbb{F}_2$ such that *for many keys* $k \in \mathbb{F}_2^\kappa$, Eq. (2.17) has many solutions $x \in \mathbb{F}_2^n$:

$$\alpha \cdot x = \beta \cdot E_k(x) + \varepsilon \quad \iff \quad \sum_{i=0}^{n-1} \alpha_i x_i = \sum_{i=0}^{n-1} \beta_i E_{k,i}(x) + \varepsilon. \tag{2.17}$$

It again distinguishes the draw of a random bijection among the block cipher $F \xleftarrow{\$} \mathcal{E}$ from the random draw among all bijections $F \xleftarrow{\$} \mathrm{Bij}(\mathbb{F}_2^n)$. However, contrary to the differential case, this is a *known plaintext/ciphertext* scenario and not a chosen-plaintext one.

2.3.4.b Linear approximations and linearity

As already mentioned, we are interested in the Boolean linear functions $f \colon \mathbb{F}_2^n \to \mathbb{F}_2$. By definition, such a function can be uniquely described by:

$$f \colon (x_0, \dots, x_{n-1}) \mapsto \sum_{i=0}^{n-1} \alpha_i x_i,$$

for some coefficients $\alpha_i \in \mathbb{F}_2$ for any i. We refer to $\alpha := (\alpha_0, \dots, \alpha_{n-1})$ as the *mask* associated to the linear function. In the context of Eqs. (2.16) and (2.17), we refer to α as the *input mask* and to β as the *output mask*. This naming is the same as the one introduced in Section 2.2.2.b for a reason. Indeed, let us denote by $Z_F^{\mathrm{lin}}(\alpha, \beta)$ the set of solutions of $\alpha \cdot x = \beta \cdot F(x)$, and $z_F^{\mathrm{lin}}(\alpha, \beta)$ its cardinality. We have in that case:

$$W_F(\alpha, \beta) = \sum_{x \in \mathbb{F}_2^n} (-1)^{\alpha \cdot x + \beta \cdot F(x)} = -\left|\mathbb{F}_2^n \setminus Z_F^{\mathrm{lin}}(\alpha, \beta)\right| + \left|Z_F^{\mathrm{lin}}(\alpha, \beta)\right| = 2 z_F^{\mathrm{lin}}(\alpha, \beta) - 2^n,$$

which proves that the (α, β) Walsh coefficient is, up to an affine transformation, the number of solutions of $\alpha \cdot x = \beta \cdot F(x)$. The LAT is therefore the right object to study resistance to linear attacks. In order to ensure that any component of F cannot be well approximated by a linear or affine function, the coefficients of the LAT of F need to be of low absolute value. The multiset of all their absolute values, as well as the maximum over them thus serves as indicators.

Definition 2.39 (Extended Walsh spectrum)**.** Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$. The extended Walsh spectrum of F is the multiset that is denoted by $\mathcal{W}(F)$ and defined by:

$$\mathcal{W}(F) := \{\!\{ |W_F(\alpha,\beta)|\,, \alpha \in \mathbb{F}_2^n, \beta \in \mathbb{F}_2^m \}\!\}.$$

▷

Remark 2.40. Despite its widespread naming, it should be noted that the extended Walsh spectrum of a function contains strictly less information than its Walsh transform, as the sign of each coefficient is omitted. ▷

Definition 2.41 (Linearity [Nyb95])**.** Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$. The linearity of F is the value denoted by $\mathcal{L}(F)$ and defined by:

$$\mathcal{L}(F) := \max_{\alpha\in\mathbb{F}_2^n, \beta\in\mathbb{F}_2^m, \beta\neq 0} |W_F(\alpha,\beta)| = \max_{\beta\in\mathbb{F}_2^n} \mathcal{L}(F_\beta).$$

▷

Example 2.42 (Linearity of the Sbox of Ascon)**.** As shown in Table 2.2, the absolute value of the Walsh coefficients of the Sbox of Ascon are all upper bounded by 16 (when the output mask is non-zero). This value is for instance reached by $W_S(\texttt{0x3},\texttt{0x1}) = 16$. Therefore, $\mathcal{L}(S) = 16$. ▷

A lower bound on linearity can easily be formulated.

Proposition 2.43 (Lower bound on linearity)**.** *Let $f\colon \mathbb{F}_2^n \to \mathbb{F}_2$. Then $\mathcal{L}(f) \geq 2^{\frac{n}{2}}$.*

Proof. Adapted from [Can16, Proposition 1.17]. First let us prove Parseval's relation, that is, $\sum_{\alpha\in\mathbb{F}_2^n} W_f(\alpha)^2 = 2^{2n}$.

$$\begin{aligned}
\sum_{\alpha\in\mathbb{F}_2^n} W_f(\alpha)^2 &= \sum_{\alpha\in\mathbb{F}_2^n}\sum_{x\in\mathbb{F}_2^n} (-1)^{f(x)+\alpha\cdot x} \sum_{y\in\mathbb{F}_2^n} (-1)^{f(y)+\alpha\cdot y} \\
&= \sum_{x\in\mathbb{F}_2^n}\sum_{y\in\mathbb{F}_2^n} (-1)^{f(x)+f(y)} \sum_{\alpha\in\mathbb{F}_2^n} (-1)^{\alpha\cdot(x+y)} \\
&= \sum_{x\in\mathbb{F}_2^n} (-1)^{f(x)+f(x)} \cdot 2^n \\
&= 2^{2n},
\end{aligned}$$

where the third equality comes from the mean of $\alpha \mapsto (x+y)\cdot\alpha$, see Lemma 2.12. Let us now assume that there exists f such that $\mathcal{L}(f) < 2^{\frac{n}{2}}$. Then f satisfies:

$$2^{2n} = \sum_{\alpha\in\mathbb{F}_2^n} W_f(\alpha)^2 \leq \sum_{\alpha\in\mathbb{F}_2^n} \mathcal{L}(f)^2 < 2^{2n},$$

which is a contradiction. □

α	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
$W_f(\alpha)$	4	4	4	-4	4	4	4	-4	4	4	4	-4	-4	-4	-4	4

Table 2.3: Walsh transform of $f\colon \mathbb{F}_2^4 \to \mathbb{F}_2, (x_0, x_1, x_2, x_3) \mapsto x_0x_1 + x_2x_3$.

This lower bound is tight as shown for instance by the function f defined by $f(x_0, x_1, x_2, x_3) = x_0x_1 + x_2x_3$, whose Walsh transform is given in Table 2.3.

An optimal function with respect to linear cryptanalysis is therefore a function that matches this bound.

Definition 2.44 (Bent function)**.** A Boolean function $f\colon \mathbb{F}_2^n \to \mathbb{F}_2$ is *bent* is a function such that $\mathcal{L}(f) = 2^{\frac{n}{2}}$. A vectorial Boolean function $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$ is bent if all its non-zero components are bent. ▷

An immediate necessary condition for a function $f\colon \mathbb{F}_2^n \to \mathbb{F}_2$ to be bent is for $\frac{n}{2}$ to be an integer, that is, for n to be even. We can also give a spectral characterization of bentness.

Lemma 2.45 (Walsh spectrum of bent functions)**.** *A function $f\colon \mathbb{F}_2^n \to \mathbb{F}_2$ is bent if and only if it satisfies: $\forall \alpha \in \mathbb{F}_2^n, \quad W_f(\alpha)^2 = 2^n$.*

Proof. The sufficiency is immediate. Regarding the necessary condition, if f is bent, for any $\alpha \in \mathbb{F}_2^n$, we have $W_f(\alpha)^2 \leq 2^n$. But we get by Parseval's relation that $\sum_{\alpha\in\mathbb{F}_2^n}(2^n - W_f(\alpha)^2) = 0$. So for this sum of positive numbers to be 0, all must be 0. □

In particular, we observe that a bent function satisfies

$$0 \neq \pm 2^{\frac{n}{2}} = W_f(0) = \sum_{x\in\mathbb{F}_2^n} (-1)^{f(x)},$$

which proves that a bent function is not balanced. This in particularly means that a vectorial bent function cannot be bijective. Indeed, for the function to take all values in $\mathbb{F}_2^n$, its coordinates must take the values 0 and 1 half of the time. Actually, we later prove in Corollary 2.55 and Proposition 2.56 that vectorial bent functions $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$ exist if and only if n is even and $2m \leq n$. Because they cannot be bijective, vectorial bent functions are not targeted when designing an Sbox used in a SPN. Nevertheless, the absolute values in the LAT must be minimized as much as possible to ensure a low linearity for the whole construction. This principle is supported by the following section.

2.3.4.c Practical linear cryptanalysis

As in the differential case, finding a highly-probable affine correlation for an intricate function $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ is impossible in practice. However, the iterated structure of symmetric primitives can also be leveraged in that case. We indeed already mentioned in Section 2.2.2.b that $\mathbf{C}^{F^{(1)} \circ F^{(0)}} = \mathbf{C}^{F^{(1)}} \mathbf{C}^{F^{(0)}}$, where $\mathbf{C}^F$ is the correlation matrix of F. This enables to express each Walsh coefficient of $F^{(1)} \circ F^{(0)}$ in terms of those of $F^{(1)}$ and $F^{(0)}$ by expressing each coordinate of a product of matrices.

Proposition 2.46 (Walsh coefficient for iterated constructions [DGV95])**.** *Let $R \geq 1$. Let $F = F^{(R-1)} \circ \ldots \circ F^{(0)}$, where $F, F^{(R-1)}, \ldots, F^{(0)}\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$. Let $\alpha^{(0)}, \alpha^{(R)} \in \mathbb{F}_2^n$. Then:*

$$\widetilde{W}_F(\alpha^{(0)}, \alpha^{(R)}) = \sum_{\alpha^{(1)}, \ldots, \alpha^{(R-1)} \in \mathbb{F}_2^n} \prod_{r=0}^{R-1} \widetilde{W}_{F^{(r)}}(\alpha^{(r)}, \alpha^{(r+1)}),$$

where $\widetilde{W}(\alpha, \beta)$ is defined (in Eq. (2.7)) by $\widetilde{W}_F(\alpha, \beta) := 2^{-n} W_F(\alpha, \beta)$ for any $\alpha, \beta \in \mathbb{F}_2^n$.

Let us now consider the key-alternating block cipher $\mathcal{E} = (E_k\colon \mathbb{F}_2^n \to \mathbb{F}_2^n)_{k \in \mathbb{F}_2^\kappa}$ such that, for any $k \in \mathbb{F}_2^\kappa$, E_k can be decomposed as follows:

$$E_k = F^{(R-1)} \circ T_{k^{(R-1)}} \circ \cdots \circ F^{(1)} \circ T_{k^{(1)}} \circ F^{(0)},$$

where each $k^{(t)}$ is derived from k through a key schedule. Then for any $k \in \mathbb{F}_2^\kappa$:

$$\widetilde{W}_{E_k}\left(\alpha^{(0)}, \alpha^{(R)}\right) = \sum_{\alpha^{(1)}, \ldots, \alpha^{(R-1)} \in \mathbb{F}_2^n} (-1)^{\sum_{r=1}^{R-1} \alpha^{(r)} \cdot k^{(r)}} \prod_{r=0}^{R-1} \widetilde{W}_{F^{(r)}}(\alpha^{(r)}, \alpha^{(r+1)}). \tag{2.18}$$

Proof. This all comes from the matrix multiplication formula applied to $\prod_{r=0}^{R-1} C_{F^{(r)}}$. The last formula is derived from the latter one by observing (thanks to Lemma 2.12) that for any $v \in \mathbb{F}_2^n$,

$$\widetilde{W}_{T_v}(\alpha, \beta) = \begin{cases} (-1)^{\beta \cdot v} & \text{if } \alpha = \beta \\ 0 & \text{otherwise.} \end{cases}$$

□

This enables an analysis based on *linear trails*. Each term of the sum in Eq. (2.18) is called the *correlation* of the associated linear trail. For a given trail, the absolute value of the correlation does not depend on the key and therefore can be obtained by studying the simple unkeyed round functions. This is made possible by the fact that the Walsh coefficients of a linear layer or of the parallel application of small Sboxes can easily be computed, in a similar way to Proposition 2.33. Furthermore, the overall correlation $\widetilde{W}_{E_k}(\alpha^{(0)}, \alpha^{(R)})$ is often approximated by only summing over one or a few *dominant* trails. However, contrary to the differential case, each term in Eq. (2.18) can be either positive or negative, and the contribution of dominant trails can be canceled out by the ignored trails. This makes the behavior of such a distinguisher hard to predict. If the key schedule is replaced by uniformly random round keys, the average over all keys for $\widetilde{W}_{E_k}(\alpha^{(0)}, \alpha^{(R)})^2$ can be expressed as the sum of squared correlations of the trails [Nyb95, Theorem 1], but its usage again implies non-trivial assumptions, in the vein of Assumptions 2.34 and 2.35, that should be manipulated with caution. We refer to the thesis of Flórez Gutiérrez [FG22] regarding key-recovery attacks based on linear distinguishers and to the one of Beyne [Bey23] for a point of view which widens linear cryptanalysis to a broad class of attacks.

2.3.4.d Linearity of functions $F\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$

Finally, when working with a function $F\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$, it is more convenient to directly work with the linear functions $f\colon \mathbb{F}_{2^n} \to \mathbb{F}_2$ to define linearity than to work with Boolean linear functions defined over $(\mathbb{F}_2)^n$. Those functions can easily be described thanks to the absolute trace.

Definition 2.47 (Trace)**.** Let $n \in \mathbb{N}$ such that $n = \ell \times k$. Let $\mathbb{L} = \mathbb{F}_{2^n}$ and $\mathbb{F} = \mathbb{F}_{2^k}$ so that $\mathbb{F} \subset \mathbb{L}$. The *trace* function defined over $\mathbb{L}$ and relative to $\mathbb{F}$ is the function $\mathrm{Tr}_{\mathbb{L}/\mathbb{F}}\colon \mathbb{L} \to \mathbb{F}$ defined by:

$$\mathrm{Tr}_{\mathbb{L}/\mathbb{F}}\colon x \mapsto \sum_{i=0}^{\ell} x^{2^{ik}}.$$

When, the domain of the trace is clear from context, we often refer to $\mathrm{Tr}_{\mathbb{L}/\mathbb{F}}$ as the trace relative to $\mathbb{F}$. Furthermore, the trace relative to $\mathbb{F}_2$ is called the *absolute trace* (of $\mathbb{L}$). ▷

The fact that the codomain of $\mathrm{Tr}_{\mathbb{L}/\mathbb{F}}$ is indeed $\mathbb{F}$ is for instance proven in the comments following [LN96, Definition 2.22]. The following proposition gathers some of the well-known properties of these functions.

Proposition 2.48 (Trace properties)**.** *Let $\mathbb{K} \subset \mathbb{F} \subset \mathbb{L}$ be a tower of extensions of finite fields of characteristic 2. Then:*

1. *($\mathbb{L}$-additivity)* $\forall x, y \in \mathbb{L}$, $\mathrm{Tr}_{\mathbb{L}/\mathbb{F}}(x + y) = \mathrm{Tr}_{\mathbb{L}/\mathbb{F}}(x) + \mathrm{Tr}_{\mathbb{L}/\mathbb{F}}(y)$.
2. *($\mathbb{F}$-scalar multiplication)* $\forall x \in \mathbb{L}, \forall \varphi \in \mathbb{F}$, $\mathrm{Tr}_{\mathbb{L}/\mathbb{F}}(\varphi x) = \varphi \mathrm{Tr}_{\mathbb{L}/\mathbb{F}}(x)$.
3. *($\mathbb{F}$-linearity)* $\mathrm{Tr}_{\mathbb{L}/\mathbb{F}}$ *is* $\mathbb{F}$*-linear.*
4. *(surjectivity)* $\mathrm{Tr}_{\mathbb{L}/\mathbb{F}}$ *is onto.*
5. *(transitivity)* $\mathrm{Tr}_{\mathbb{L}/\mathbb{K}} = \mathrm{Tr}_{\mathbb{F}/\mathbb{K}} \circ \mathrm{Tr}_{\mathbb{L}/\mathbb{F}}$.

Proof. We refer to [LN96, Theorems 2.23 & 2.26] for the proof of those results. □

Proposition 2.49 (Trace and linear functions). *Let $n = \ell k$ and $F\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^k}$ be a linear function. Then, there exists a unique $\alpha \in \mathbb{F}_{2^n}$ such that F can be described as $F\colon x \mapsto \mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\alpha x)$.*

Proof. Adapted from [LN96, Theorem 2.24]. Because multiplication by a constant and the trace are linear, their composition $x \mapsto \mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\alpha x)$ is $\mathbb{F}_{2^k}$-linear, for any $\alpha \in \mathbb{F}_{2^n}$. Let $\alpha, \beta \in \mathbb{F}_{2^n}, \alpha \neq \beta$. By surjectivity of $\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}$, there exists $y \in \mathbb{L}$ such that $\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(y) \neq 0$. Let us denote by z the value $z := y(\alpha+\beta)^{-1}$, we therefore observe that

$$0 \neq \mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}\left((\alpha+\beta)z\right) = \mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}\left(\alpha z\right) + \mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}\left(\beta z\right),$$

where we used $\mathbb{L}$-additivity. This proves that $x \mapsto \mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\alpha x)$ and $x \mapsto \mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\beta x)$ are distinct. There are therefore 2^n such functions among the $(2^k)^\ell = 2^n$ linear functions from $\mathbb{L}$ to $\mathbb{F}_2$ (2^k choices for each of the ℓ elements of an $\mathbb{F}_{2^k}$-basis of $\mathbb{F}_{2^n}$), which proves the announced statement. □

Therefore, for $F\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$, we rather look at equations of the form:

$$\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_2}(\alpha x) = \mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_2}(\beta F(x)) + \varepsilon, \tag{2.19}$$

where $\alpha, \beta, x \in \mathbb{F}_{2^n}$. It is therefore convenient to overload the notation of Walsh coefficients, so that for any $\alpha, \beta \in \mathbb{F}_{2^n}$, $W_F(\alpha,\beta)$ is defined by:

$$W_F(\alpha,\beta) := \sum_{x \in \mathbb{F}_{2^n}} (-1)^{\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_2}(\alpha x) + \mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_2}(\beta F(x))}.$$

This way, $\mathcal{L}(F)$ can still be computed as the maximum over all Walsh coefficients:

$$\mathcal{L}(F) = \max_{\alpha,\beta \in \mathbb{F}_{2^n}, \beta \neq 0} \left|W_F(\alpha,\beta)\right|.$$

2.3.5 Link between linear and differential cryptanalysis

While the approaches of linear and differential cryptanalysis are rather different, both techniques remain very close [CV95, BN13]. In this section, we explain in the case of a function $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ how to compute any coefficient of its DDT from the squared coefficients of its LAT, and *vice versa*. We also point out that for Boolean functions, the notion of bentness and perfect nonlinearity coincide.

2.3.5.a Squared Walsh transform and difference distribution table

Proposition 2.50 (Link between LAT and DDT [CV95]). *Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$. Then:*

$$\forall \alpha, \beta \in \mathbb{F}_2^n, \quad W_F(\alpha,\beta)^2 = \sum_{\Delta^{\mathrm{in}} \in \mathbb{F}_2^n} \sum_{\Delta^{\mathrm{out}} \in \mathbb{F}_2^n} (-1)^{\alpha\cdot\Delta^{\mathrm{in}} + \beta\cdot\Delta^{\mathrm{out}}} \delta_F(\Delta^{\mathrm{in}}, \Delta^{\mathrm{out}}).$$

Conversely, it holds that:

$$\forall \Delta^{\mathrm{in}}, \Delta^{\mathrm{out}} \in \mathbb{F}_2^n, \quad \delta_F(\Delta^{\mathrm{in}}, \Delta^{\mathrm{out}}) = 2^{-2n} \sum_{\alpha \in \mathbb{F}_2^n} \sum_{\beta \in \mathbb{F}_2^n} (-1)^{\alpha\cdot\Delta^{\mathrm{in}} + \beta\cdot\Delta^{\mathrm{out}}} W_F(\alpha,\beta)^2.$$

Proof. Adapted from [Can16]. We start by proving the second statement. Let $\Delta^{\text{in}}, \Delta^{\text{out}} \in \mathbb{F}_2^n$. It holds that:

$$\begin{aligned}\delta_F(\Delta^{\text{in}}, \Delta^{\text{out}}) &= \left|\left\{x \in \mathbb{F}_2^n, F(x+\Delta^{\text{in}}) + F(x) = \Delta^{\text{out}}\right\}\right| \\ &= \left|\left\{(x,y) \in (\mathbb{F}_2^n)^2, y = x + \Delta^{\text{in}}, F(x) + F(y) = \Delta^{\text{out}}\right\}\right| \\ &= \sum_{x\in\mathbb{F}_2^n}\sum_{y\in\mathbb{F}_2^n} \mathbf{1}_0\left(x+y+\Delta^{\text{in}}\right)\mathbf{1}_0\left(F(x)+F(y)+\Delta^{\text{out}}\right).\end{aligned}$$

But for any $x, y \in \mathbb{F}_2^n$, because of Lemma 2.12, we have:

$$\begin{aligned}\mathbf{1}_0\left(x+y+\Delta^{\text{in}}\right) &= 2^{-n}\sum_{\alpha\in\mathbb{F}_2^n}(-1)^{\alpha\cdot(x+y+\Delta^{\text{in}})} \\ \mathbf{1}_0\left(F(x)+F(y)+\Delta^{\text{out}}\right) &= 2^{-n}\sum_{\beta\in\mathbb{F}_2^n}(-1)^{\beta\cdot(F(x)+F(y)+\Delta^{\text{out}})}.\end{aligned}$$

We then obtain:

$$\begin{aligned}\delta_F(\Delta^{\text{in}}, \Delta^{\text{out}}) &= 2^{-2n}\sum_{x\in\mathbb{F}_2^n}\sum_{y\in\mathbb{F}_2^n}\sum_{\alpha\in\mathbb{F}_2^n}(-1)^{\alpha\cdot(x+y+\Delta^{\text{in}})}\sum_{\beta\in\mathbb{F}_2^n}(-1)^{\beta\cdot(F(x)+F(y)+\Delta^{\text{out}})} \\ &= 2^{-2n}\sum_{\alpha\in\mathbb{F}_2^n}\sum_{\beta\in\mathbb{F}_2^n}(-1)^{\alpha\cdot\Delta^{\text{in}}+\beta\cdot\Delta^{\text{out}}}\sum_{x\in\mathbb{F}_2^n}(-1)^{\alpha\cdot x+\beta\cdot F(x)}\sum_{y\in\mathbb{F}_2^n}(-1)^{\alpha\cdot y+\beta\cdot F(y)} \\ &= 2^{-2n}\sum_{\alpha\in\mathbb{F}_2^n}\sum_{\beta\in\mathbb{F}_2^n}(-1)^{\alpha\cdot\Delta^{\text{in}}+\beta\cdot\Delta^{\text{out}}}W_F(\alpha,\beta)^2.\end{aligned}$$

In other words, we proved that the functions $G, H \colon (\mathbb{F}_2^n)^2 \to \mathbb{N}$ defined by:

$$G\colon (\alpha,\beta) \mapsto W_F(\alpha,\beta)^2, \quad H\colon (\Delta^{\text{in}}, \Delta^{\text{out}}) \mapsto \delta_F(\Delta^{\text{in}}, \Delta^{\text{out}})$$

actually satisfy $H = \widehat{G}$, where $\widehat{\cdot}$ is defined as in Proposition 2.13, but for functions with $2n$ variables. According to Proposition 2.14, it therefore holds that $2^{-2n}G = \widehat{H}$, which proves the first statement. □

In other words, the DDT of a function $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ contains as much information as the squared Walsh coefficients of F. In particular, the DDT contains strictly less information than the LAT because the sign of each Walsh coefficient cannot be recovered from the DDT. Contrary to the case of the Walsh transform, two distinct functions can then have an identical DDT.

2.3.5.b Vectorial bentness and perfect nonlinearity

This close relationship between linear and differential cryptanalysis can also be highlighted by the fact perfect nonlinearity and bentness, which are respectively defined at the end of Section 2.3.3.c and in Definition 2.44, actually coincide in the case of Boolean functions.

First, we give a differential characterization of bentness.

Lemma 2.51 (Derivatives of bent functions)**.** *Let n be even, and $f\colon \mathbb{F}_2^n \to \mathbb{F}_2$. Then f is a bent function if and only if, for any $\Delta \in \mathbb{F}_2^n \setminus \{0\}$, $D_\Delta f$ is balanced.*

Proof. This comes from a similar reasoning than in the proof of Proposition 2.50. Indeed, for any $\alpha \in \mathbb{F}_2^n$:

$$\begin{aligned} W_f(\alpha)^2 &= \sum_{x\in\mathbb{F}_2^n} (-1)^{\alpha\cdot x+f(x)} \sum_{y\in\mathbb{F}_2^n} (-1)^{\alpha\cdot y+f(y)} \\ &= \sum_{x\in\mathbb{F}_2^n} (-1)^{\alpha\cdot x} \sum_{\Delta\in\mathbb{F}_2^n} (-1)^{\alpha\cdot(\Delta+x)+f(x+\Delta)+f(x)} \\ &= \sum_{\Delta\in\mathbb{F}_2^n} (-1)^{\alpha\cdot\Delta} W_{D_\Delta f}(0), \end{aligned} \tag{2.20}$$

where, for any fixed x, we use $\Delta \leftarrow y+x$ as change of variables to obtain the second equality. In particular, if all $D_\Delta f$ are balanced for $\Delta \neq 0$, then $W_{D_\Delta f}(0) = 0$ for all $\Delta \neq 0$ and therefore for any $\alpha \in \mathbb{F}_2^n$, we have:

$$W_f(\alpha)^2 = (-1)^{\alpha\cdot 0} W_{D_0 f}(0) = 2^n,$$

so that f is bent. Conversely, by Fourier inversion, it holds that:

$$\forall\ \Delta \in \mathbb{F}_2^n, \quad W_{D_\Delta f}(0) = 2^{-n} \sum_{\alpha\in\mathbb{F}_2^n} (-1)^{\alpha\cdot\Delta} W_f(\alpha)^2.$$

Therefore if $W_f(\alpha)^2 = 2^n$ for any α, then we have:

$$\forall\ \Delta \in \mathbb{F}_2^n, \quad W_{D_\Delta f}(0) = \sum_{\alpha\in\mathbb{F}_2^n} (-1)^{\alpha\cdot\Delta} = 2^n \mathbf{1}_0(\Delta),$$

and $D_\Delta f$ is balanced for any $\Delta \neq 0$. □

The following definition and lemma are needed to link bentness and perfect nonlinearity.

Definition 2.52 (Balanced vectorial function)**.** Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$. The function F is *balanced* if for any $y \in \mathbb{F}_2^m$, the preimage $F^{-1}(\{y\})$ has cardinality 2^{n-m}. ▷

Lemma 2.53. *Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$. The function F is balanced if and only if all its non-zero components $\alpha \cdot F$, with $\alpha \neq 0$ are balanced.*

Proof. Adapted from [Car21, Proposition 35]. Because of Lemma 2.12, we observe that for $x \in \mathbb{F}_2^n, y \in \mathbb{F}_2^m$, it holds that:

$$\sum_{v\in\mathbb{F}_2^m} (-1)^{v\cdot(F(x)+y)} = 2^m \mathbf{1}_y(F(x)).$$

Therefore, we observe that, for any $y \in \mathbb{F}_2^m$, it holds that:

$$\begin{aligned} \left|F^{-1}(\{y\})\right| &= \sum_{x\in\mathbb{F}_2^n} \mathbf{1}_y(F(x)) \\ &= 2^{-m} \sum_{x\in\mathbb{F}_2^n} \sum_{v\in\mathbb{F}_2^m} (-1)^{v\cdot(F(x)+y)} \\ &= 2^{-m} \sum_{v\in\mathbb{F}_2^m} (-1)^{v\cdot y} \sum_{x\in\mathbb{F}_2^n} (-1)^{v\cdot F(x)}. \\ &= 2^{-m} \sum_{v\in\mathbb{F}_2^m} (-1)^{v\cdot y} W_F(0,v). \end{aligned}$$

Therefore, if for any $v \neq 0$, $v \cdot F$ is balanced then $W_F(0,v) = 0$ for any $v \neq 0$ and therefore:

$$\forall\ y \in \mathbb{F}_2^n, \quad \left|F^{-1}(\{y\})\right| = 2^{-m}(-1)^{0\cdot y} W_F(0,0) = 2^{n-m}.$$

Conversely, it holds by Fourier transform that:

$$\forall v \in \mathbb{F}_2^m, \quad W_F(0,v) = \sum_{y\in\mathbb{F}_2^m} (-1)^{y\cdot v} \left|F^{-1}(\{y\})\right|.$$

If all $\left|F^{-1}(\{y\})\right|$ are equal to 2^{n-m}, then:

$$\forall v \in \mathbb{F}_2^m, \quad W_F(0,v) = 2^{n-m} \sum_{y\in\mathbb{F}_2^m} (-1)^{y\cdot v} = 2^{n-m}2^m \mathbf{1}_0(v) = 2^n \mathbf{1}_0(v).$$

□

Corollary 2.54 (Bentness and perfect nonlinearity [Nyb91]). *Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$. Then F is bent if and only if F is perfect nonlinear.*

Proof. Adapted from [Car21, Section 6.4]. We first state equivalent definitions for both perfect nonlinearity and bentness, and then prove the equivalence between these two new formulations.

Recall that F is perfect nonlinear if and only if $\delta_F = 2^{n-m}$. This implies that for any $\Delta^{\text{in}} \in \mathbb{F}_2^n \setminus \{0\}, \Delta^{\text{out}} \in \mathbb{F}_2^m$, $2^{n-m} \leq \delta_F(\Delta^{\text{in}}, \Delta^{\text{out}}) \leq 2^{n-m}$, *i.e.* $\delta_F(\Delta^{\text{in}}, \Delta^{\text{out}}) = 2^{n-m}$. In other words, F is perfect non-linear if for any $\Delta \neq 0$, $D_\Delta F$ is balanced.

Furthermore, because of Lemma 2.51, a vectorial function F is bent if and only for any $\Delta \neq 0, v \neq 0$, $D_\Delta(v \cdot F) = v \cdot D_\Delta F$ is balanced.

By applying Lemma 2.53 to all derivatives $D_\Delta F$ with $\Delta \neq 0$, we conclude that the definitions of bentness and perfect nonlinearity coincide. □

Corollary 2.55 (Necessary condition for the existence of perfect nonlinear functions [Nyb91]). *Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$. If F is perfect nonlinear then n is even and $2m \leq n$.*

Proof. Adapted from [CV95, Theorem 3]. We prove it using the bentness of F. The fact that n is even is mentioned just after Definition 2.44. Recall from the proof of Lemma 2.53 that for any function $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$ and any $y \in \mathbb{F}_2^m$, it holds that:

$$\left|F^{-1}(\{y\})\right| = 2^{-m} \sum_{v\in\mathbb{F}_2^m} (-1)^{v\cdot y} W_F(0,v).$$

Here F is bent so we further obtain:

$$\left|F^{-1}(\{y\})\right| = 2^{n-m} + 2^{\frac{n}{2}-m} \sum_{v\in\mathbb{F}_2^m\setminus\{0\}} (-1)^{v\cdot y+\varepsilon_v} = 2^{\frac{n}{2}-m}\left(2^{\frac{n}{2}} + \sum_{v\in\mathbb{F}_2^m\setminus\{0\}} (-1)^{v\cdot y+\varepsilon_v}\right),$$

where $\varepsilon_v \in \mathbb{F}_2$ is given by the sign of the Walsh coefficient $W_F(0,v)$. The term $\sigma := \sum_{v\in\mathbb{F}_2^m\setminus\{0\}} (-1)^{v\cdot y+\varepsilon_v}$ is a sum with an odd number of terms which are all odd, so σ is odd itself. Therefore $2^{\frac{n}{2}} + \sigma$ is an odd integer and in particular it is not 0. If we suppose that $\frac{n}{2} - m < 0$, then $2^{\frac{n}{2}-m} < 1$, so the cardinality $\left|F^{-1}(\{y\})\right|$ cannot be an integer itself. This implies that n, m must satisfy $\frac{n}{2} \geq m$. □

As shown in the following proposition, this necessary condition on the values of n and m for the existence of perfect nonlinear functions is actually a sufficient one.

Proposition 2.56 (Maiorana-McFarland construction [McF73]). *Let $H\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ and let $G\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ be a bijection. Let $F\colon (\mathbb{F}_{2^n})^2 \to \mathbb{F}_{2^n}$ be defined by:*

$$F\colon (x,y) \mapsto xG(y) + H(y).$$

Then F is a bent function.

Proof. Let $\alpha = (\alpha_0, \alpha_1) \in (\mathbb{F}_{2^n})^2$. Let $\beta \in \mathbb{F}_{2^n} \setminus \{0\}$. Then it holds that:

$$\begin{aligned} W_F(\alpha,\beta) &= \sum_{x\in\mathbb{F}_{2^n}} \sum_{y\in\mathbb{F}_{2^n}} (-1)^{\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_2}(\alpha_0 x+\alpha_1 y+\beta(xG(y)+H(y)))} \\ &= \sum_{y\in\mathbb{F}_{2^n}} (-1)^{\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_2}(\alpha_1 y+\beta H(y))} \sum_{x\in\mathbb{F}_{2^n}} (-1)^{\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_2}(x(\alpha_0+\beta G(y)))} \\ &= 2^n \sum_{y\in\mathbb{F}_{2^n}} (-1)^{\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_2}(\alpha_1 y+\beta H(y))} \mathbf{1}_0(\alpha_0 + \beta G(y)). \end{aligned}$$

But as G is bijective and $\beta \neq 0$, we observe that $\alpha_0 + \beta G(y) = 0$ is equivalent to $y = G^{-1}(\alpha_0\beta^{-1})$. Therefore the previous sum contains a single non-zero term which is the one corresponding to $y = G^{-1}(\alpha_0\beta^{-1})$. In other words, it holds that:

$$W_F(\alpha,\beta) = 2^n(-1)^{\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_2}\left(\alpha_1 G^{-1}(\alpha_0\beta^{-1})+\beta H(G^{-1}(\alpha_0\beta^{-1}))\right)}.$$

In particular we observe that $|W_F(\alpha,\beta)| = 2^n$, and therefore F is a bent function. □

Example 2.57. Let $n \geq 1$. Then the function $F\colon (\mathbb{F}_{2^n})^2 \to \mathbb{F}_{2^n}$ defined by:

$$F\colon (x, y) \mapsto xy$$

is a bent function. ▷

Proposition 2.56 proves the existence of bent functions for the case $n = 2m$. The existence in the case where $m < \frac{n}{2}$ can for instance be deduced by only keeping m coordinates of a Maiorana-McFarland function $F\colon (\mathbb{F}_2^n)^2 \to \mathbb{F}_2^n$.

2.4 Equivalence relations

With those cryptographic criteria in mind, finding functions that satisfy optimal resistance, for instance, to linear and differential attacks is one of the biggest challenges. In order to effectively tackle this problem, it is necessary to study functions up to equivalence. This not only enables us to classify vectorial Boolean functions, but it also helps building effective algorithms to search for optimal objects. This section therefore presents the main equivalence relations used for studying vectorial Boolean functions. These equivalence relations hold for functions $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$ where n and m can be distinct, but for the sake of uniformity with the remaining of this manuscript, we restrict this presentation to the case $n = m$. The more general definitions and statements can easily be adapted from this specific case.

Definition 2.58 (Affine equivalence)**.** Let $F, G\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$. Then, F and G are said to be *affine equivalent* if there exist two affine bijective mappings $A, B\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ such that:

$$G = A \circ F \circ B.$$

It is denoted by $F \sim_{\text{aff}} G$. If there exist such A and B that are instead linear, F and G are said to be *linearly equivalent*, which is denoted by $F \sim_{\text{lin}} G$. ▷

Definition 2.59 (Extended affine equivalence (EA))**.** Let $F, G\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$. Then, F and G are said to be *extended affine equivalent* if there exist two affine bijective mappings $A, B\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ and an affine or constant mapping $C\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ such that:

$$G = A \circ F \circ B + C. \tag{2.21}$$

It is denoted by by $F \sim_{\text{EA}} G$. ▷

Definition 2.60 (Carlet-Charpin-Zinoviev equivalence (CCZ) [CCZ98, BCP06])**.** Let $F, G\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$. Then, F and G are said to be *CCZ equivalent* if there exists an affine bijective mapping $\mathcal{A}\colon (\mathbb{F}_2^n)^2 \to (\mathbb{F}_2^n)^2$ such that:

$$\mathcal{G}_G = \mathcal{A}(\mathcal{G}_F),$$

where $\mathcal{G}_F$ is the graph of F: $\mathcal{G}_F := \{(x, F(x)), x \in \mathbb{F}_2^n\}$. It is denoted by $F \sim_{\text{CCZ}} G$. ▷

More generally, as defined and characterized in the following definition and lemma, the *admissible mappings* for F are the affine mappings that lead to a CCZ-equivalence relation.

Definition 2.61 (Admissible mapping [CP19, Definition 4]). Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$. Let $\mathcal{A}$ be an affine mapping over $\mathbb{F}_2^{2n}$. The mapping $\mathcal{A}$ is *admissible for* F if $\mathcal{A}(\mathcal{G}_F)$ is the graph of a function, *i.e.* if there exists $G\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ such that $\mathcal{A}(\mathcal{G}_F) = \mathcal{G}_G$. ▷

Lemma 2.62 (Characterization of an admissible mapping [CP19, Definition 4]). *Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$. Let $\mathcal{A}$ be an affine mapping over $\mathbb{F}_2^{2n}$ that we decompose as:*

$$\mathcal{A}(x,y) = (\mathcal{A}_1(x,y), \mathcal{A}_2(x,y)),$$

where $\mathcal{A}_1, \mathcal{A}_2\colon \mathbb{F}_2^n \times \mathbb{F}_2^n \to \mathbb{F}_2^n$. Then $\mathcal{A}$ is admissible for F if and only if $x \mapsto \mathcal{A}_1(x, F(x))$ is bijective.

Proof. It is clear that $\mathcal{A}(\mathcal{G}_F) = \{(\mathcal{A}_1(x,F(x)), \mathcal{A}_2(x,F(x))), x \in \mathbb{F}_2^n\}$. For this set to be the graph of a function over $\mathbb{F}_2^n$, it is necessary that the first coordinate takes once, and only once, each possible value, or in other words that $\{\mathcal{A}_1(x,F(x)), x \in \mathbb{F}_2^n\} = \mathbb{F}_2^n$. This is equivalent to stating that $x \mapsto \mathcal{A}_1(x,F(x))$ is bijective. Conversely, if the first coordinate of $\mathcal{A}(\mathcal{G}_F)$ takes all values a single time, then the function $G\colon x \mapsto y$ where y is the only value such that $(x,y) \in \mathcal{A}(\mathcal{G}_F)$ is well-defined. By construction G satisfies $\mathcal{A}(\mathcal{G}_F) = \mathcal{G}_G$. □

Naturally, we can sort those equivalence relations from the most restrictive to the most general one.

Proposition 2.63 (Partitions of equivalence classes into smaller classes). *Let $F, G\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$.*

- *If $F \sim_{\text{lin}} G$, then $F \sim_{\text{aff}} G$.*
- *If $F \sim_{\text{aff}} G$, then $F \sim_{\text{EA}} G$.*
- *If $F \sim_{\text{EA}} G$, then $F \sim_{\text{CCZ}} G$.*

Before proving the previous proposition, we introduce matrix-like notation for affine mappings. If $A_0, \dots, A_3\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ are affine mappings, the affine mapping $\begin{pmatrix} A_0 & A_1 \\ A_2 & A_3 \end{pmatrix} : (\mathbb{F}_2^n)^2 \to (\mathbb{F}_2^n)^2$ is defined by :

$$\begin{pmatrix} A_0 & A_1 \\ A_2 & A_3 \end{pmatrix} : \begin{pmatrix} x \\ y \end{pmatrix} \to \begin{pmatrix} A_0(x) + A_1(y) \\ A_2(x) + A_3(y) \end{pmatrix} = \begin{pmatrix} L_0 & L_1 \\ L_2 & L_3 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} a_0 + a_1 \\ a_2 + a_3 \end{pmatrix},$$

where for any i, $L_i\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ is the *linear part* of A_i, *i.e.* $L_i := A_i + A_i(0)$, and $a_i := A_i(0)$ its *constant term*. By analogy with matrices, we also omit the composition symbol $\circ$ when composing two affine mappings together.

Proof. The first two statements are immediate corollaries of the definitions, so we only focus on the last one. Given mappings F, G, A, B satisfying Eq. (2.21), we observe that for any $x \in \mathbb{F}_2^n$:

$$\begin{pmatrix} B^{-1} & 0 \\ CB^{-1} & A \end{pmatrix} \begin{pmatrix} x \\ F(x) \end{pmatrix} = \begin{pmatrix} B^{-1}(x) \\ G \circ B^{-1}(x) \end{pmatrix},$$

because $G = A \circ F \circ B + C \iff A \circ F = G \circ B^{-1} + CB^{-1}$. But B^{-1} being bijective, this implies that:

$$\begin{pmatrix} B^{-1} & 0 \\ CB^{-1} & A \end{pmatrix} \mathcal{G}_F = \mathcal{G}_G.$$

□

In particular, EA equivalence corresponds to CCZ equivalence with lower-triangular mappings, while affine equivalence corresponds to CCZ equivalence with diagonal mappings. Finally, let us mention the cryptographic properties that remain invariant within an equivalence class.

Proposition 2.64 (Invariants of equivalence classes)**.** *Let $F, G \colon \mathbb{F}_2^n \to \mathbb{F}_2^n$.*

- *If $\deg_a(F) > 1$ and $F \sim_{\mathrm{EA}} G$, then $\deg_a(F) = \deg_a(G)$.*
- *If $F \sim_{\mathrm{CCZ}} G$ then the number of solutions of differential (resp. linear) equations of F and G are in one-to-one correspondence. More precisely, let $\mathcal{G}_G = \mathcal{A}(\mathcal{G}_F)$ where $\mathcal{A} = \mathcal{L} + c$, with $\mathcal{L} \colon (\mathbb{F}_2^n)^2 \to (\mathbb{F}_2^n)^2$, and $c \in \mathbb{F}_2^n$. Then, for any $\alpha, \beta, \Delta^{\mathrm{in}}, \Delta^{\mathrm{out}} \in \mathbb{F}_2^n$:*

$$\delta_G(\Delta^{\mathrm{in}}, \Delta^{\mathrm{out}}) = \delta_F\left(\mathcal{L}^{-1}(\Delta^{\mathrm{in}}, \Delta^{\mathrm{out}})\right) \quad \textit{and} \tag{2.22}$$

$$W_G(\alpha, \beta) = (-1)^{c \cdot (\alpha, \beta)} W_F\left(\mathcal{L}^\top(\alpha, \beta)\right). \tag{2.23}$$

Proof. First, we observe that pre- and post-composition with an affine mapping keep the degree unchanged. This is also the case of addition with an affine or constant mapping unless the function is itself affine or constant. Indeed, any affine or constant function F is equivalent to the zero function because $F + F = 0$, and the class of the zero function is therefore made of all functions of degree 0 and 1.

Regarding the second statement, let us decompose $\mathcal{L}$ as $\mathcal{L} = \begin{pmatrix} A & B \\ C & D \end{pmatrix}$ and c as $c = (a, b)$. Because $\mathcal{G}_G = \mathcal{A}(\mathcal{G}_F)$, enumerating all pairs $(y, G(y))$ comes back to enumerating all pairs $\mathcal{L}(x, F(x))$. Therefore, we obtain:

$$\begin{aligned}
W_G(\alpha, \beta) &= \sum_{y \in \mathbb{F}_2^n} (-1)^{\alpha \cdot y + \beta \cdot G(y)} \\
&= \sum_{x \in \mathbb{F}_2^n} (-1)^{\alpha \cdot (Ax + BF(x) + a) + \beta \cdot (Cx + DF(x) + b)} \\
&= (-1)^{\alpha \cdot a + \beta \cdot b} \sum_{x \in \mathbb{F}_2^n} (-1)^{\left(A^\top \alpha + C^\top \beta\right) \cdot x + \left(B^\top \alpha + D^\top \beta\right) \cdot F(x)} \\
&= (-1)^{c \cdot (\alpha, \beta)} W_F(\mathcal{L}^\top(\alpha, \beta)),
\end{aligned}$$

where the last equality comes from the fact that:

$$\mathcal{L}^\top = \begin{pmatrix} A & B \\ C & D \end{pmatrix}^\top = \begin{pmatrix} A^\top & C^\top \\ B^\top & D^\top \end{pmatrix}.$$

In the same way, we observe that:

$$\begin{aligned}
\delta_G(\Delta^{\text{in}}, \Delta^{\text{out}}) &= \left| \left\{ y \in \mathbb{F}_2^n, G(y) + G(y + \Delta^{\text{in}}) = \Delta^{\text{out}} \right\} \right| \\
&= \left| \left\{ (y, z) \in (\mathbb{F}_2^n)^2, y + z = \Delta^{\text{in}}, G(y) + G(z) = \Delta^{\text{out}} \right\} \right| \\
&= \left| \left\{ (x, w) \in (\mathbb{F}_2^n)^2, \begin{array}{l} A(x+z) + B(F(x) + F(z)) = \Delta^{\text{in}}, \\ C(x+z) + D(F(x) + F(z)) = \Delta^{\text{out}} \end{array} \right\} \right| \\
&= \left| \left\{ (x, w) \in (\mathbb{F}_2^n)^2, \mathcal{L} \begin{pmatrix} x + z \\ F(x) + F(z) \end{pmatrix} = \begin{pmatrix} \Delta^{\text{in}} \\ \Delta^{\text{out}} \end{pmatrix} \right\} \right| \\
&= \left| \left\{ (x, w) \in (\mathbb{F}_2^n)^2, \begin{pmatrix} x + z \\ F(x) + F(z) \end{pmatrix} = \mathcal{L}^{-1} \begin{pmatrix} \Delta^{\text{in}} \\ \Delta^{\text{out}} \end{pmatrix} \right\} \right| \\
&= \delta_F \left(\mathcal{L}^{-1}(\Delta^{\text{in}}, \Delta^{\text{out}}) \right),
\end{aligned}$$

where the last equality comes from the same reasoning that leads to the first two equalities, but the other way around. □

Corollary 2.65 (CCZ equivalence, linearity, differential uniformity)**.** *Let $F, G \colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ be such that $F \sim_{\text{CCZ}} G$. Then F and G share the same extended Walsh spectrum and* differential spectrum. *In other words, the following equalities between* multisets *holds:*

$$\mathcal{W}(F) = \mathcal{W}(G), \ \textit{and}$$

$$\{\!\{\delta_F(\Delta^{\text{in}}, \Delta^{\text{out}}), \Delta^{\text{in}}, \Delta^{\text{out}} \in \mathbb{F}_2^n\}\!\} = \{\!\{\delta_G(\Delta^{\text{in}}, \Delta^{\text{out}}), \Delta^{\text{in}}, \Delta^{\text{out}} \in \mathbb{F}_2^n\}\!\}.$$

In particular, F and G share the same differential uniformity and linearity:

$$\delta_F = \delta_G \quad \textit{and} \quad \mathcal{L}(F) = \mathcal{L}(G),$$

where the extended Walsh spectrum $\mathcal{W}$ is defined in Definition 2.39.

CCZ equivalence is (so far) the least restrictive equivalence relation that preserves linearity and differential uniformity. This is also the case for EA equivalence with respect to algebraic degree. The following counter-example proves that inversion is a particular case of CCZ equivalence, and that CCZ equivalence does not preserve the algebraic degree.

Example 2.66 (CCZ equivalence and algebraic degree)**.** Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ be a bijective mapping. We easily observe that the graphs of F and F^{-1} satisfy the equation :

$$\begin{pmatrix} 0 & \mathrm{I_n} \\ \mathrm{I_n} & 0 \end{pmatrix} \mathcal{G}_F = \mathcal{G}_{F^{-1}}.$$

So a function and its inverse lie in the same CCZ equivalence class. However in general, F^{-1} does not have the same algebraic degree as F. For instance, $F\colon \mathbb{F}_{2^5} \to \mathbb{F}_{2^5}, x \mapsto x^3$ is bijective because $\gcd(3, 2^5-1) = 1$. Its inverse function is $F^{-1}\colon x \mapsto x^{21}$ since $3 \times 21 \equiv 63 \equiv 1 \bmod 31$. But $\mathrm{wt}(3) = 2 \neq 3 = \mathrm{wt}(21)$. So in general, F and F^{-1} do not lie in the same EA class. ▷

Remark 2.67. The particular case of CCZ equivalence that corresponds to functional inversion was generalized by Canteaut & Perrin [CP19] into the so-called *function twisting*. It is proven in [CP19, Theorem 3] that CCZ equivalence can be fully described thanks to extended-affine and twist equivalences. ▷

Finally, the following example highlights the relevancy of extended affine equivalence in the context of the design of cryptographic primitives.

Example 2.68 (Sboxes of Ascon and SHA3)**.** As we can see on Figure 2.4, which is extracted from [Dob+21], the Sbox S of Ascon can be computed as the composition of three functions $L_{\mathrm{in}}, \chi, L_{\mathrm{out}}\colon \mathbb{F}_2^5 \to \mathbb{F}_2^5$ whose ANF are defined by:

$$L_{\mathrm{in}} \begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} := \begin{pmatrix} x_0 + x_1 \\ x_1 \\ x_2 + x_3 \\ x_3 \\ x_4 + x_0 \end{pmatrix}, \quad L_{\mathrm{out}} \begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} := \begin{pmatrix} x_0 \\ x_1 + x_2 \\ x_2 + 1 \\ x_3 + x_4 \\ x_0 + x_4 \end{pmatrix},$$

$$\chi \begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} := \begin{pmatrix} x_0 + (x_4+1)x_3 \\ x_1 + (x_0+1)x_4 \\ x_2 + (x_1+1)x_0 \\ x_3 + (x_2+1)x_1 \\ x_4 + (x_3+1)x_2 \end{pmatrix}.$$

In other words, it holds that:

$$S = L_{\mathrm{out}} \circ \chi \circ L_{\mathrm{in}}.$$

From the ANF of L_{in} and L_{out}, we easily observe that they are affine bijective mappings. This implies that S and χ are affine-equivalent. The Sbox χ is actually

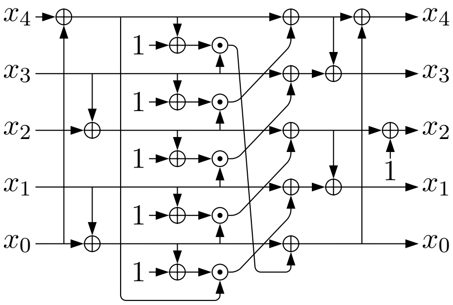


Figure 2.4: A possible implementation of the Sbox of Ascon extracted from [Dob+21].

the Sbox used by the cryptographic permutation Keccak−f which is a component of the standardized hash function SHA3 [Nis07, Ber+11].

▷

Chapter 3

Higher-order cryptanalysis and its application to Ascon

Lightweight cryptography, that is introduced in Section 1.3.2.a promotes different trade-offs between security, performances and cost than the ones proposed by classical algorithms such as the AES. This pushes designers to build ciphers which should be implemented very efficiently, but without cutting back security. Still, some strong design choices need to be made in order to remain competitive.

Ascon [Dob+21], which is both one of the winners of CAESAR [Cae13] and the winner of the recent NIST standardization process [Nis17], is indisputably the lightweight cipher that attracts the most attention today. One of the most notable decisions made by its designers, Dobraunig, Eichlseder, Mendel & Schläffer, is the choice of a sparse quadratic 5-bit Sbox, which serves as example all along Chapter 2. This quadraticity prevents the algebraic degree of the whole construction from increasing rapidly, and the sparsity prevents the ANF from quickly becoming intricate. As noticed in Section 2.3.1, this may make the cipher more vulnerable to algebraic attacks.

Among them, higher-order differential [Knu95, Lai94] attacks take advantage of both properties in a variety of forms. Cube testers [Aum+09], or integral attacks [KW02] leverage the fact that a specific known monomial cannot appear in the ANF of the targeted cryptographic function. This enables an adversary to distinguish such a function from a random one for which any monomial is expected to appear in each coordinate with probability $\frac{1}{2}$. On the other hand, the goal of a cube attack [DS09] or methods based on the division property [Tod15b] is to find simple equations in key bits by targeting some specific coefficients in the ANF. All those techniques correspond to the same theory, but viewed through different lenses.

The goal of this chapter is to first present the general framework of higher-order differential attacks by relying on the prerequisite from Chapter 2. Then, a precise description of Ascon is given, as well as a literature review of previous higher-order differential attacks against it. Subsequently, we describe a new higher-order differential attack against Ascon which, when the attacker is given sufficient power, breaks its confidentiality, but not the claim made by the designers. Still, this attack leads to a deeper understanding of the inner components of Ascon, especially its Sbox. With the benefit of hindsight, we conclude this chapter with general comments on higher-order differential attacks. This chapter is based on a joint work with Anne Canteaut & Léo Perrin that is published in the IACR Transactions on Symmetric Cryptology, 2022(4) [BCP22].

Contents

3.1 Higher-order differential cryptanalysis

3.1.1 Higher-order derivatives

The notion of *higher-order derivative* is the natural generalization of Definition 2.29 to successive derivations along distinct directions. In that case, the order in which the different derivations are made does not matter.

Lemma 3.1 (Multiple derivations formula [Lai94, Propositions 3 & 4])**.** *Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$. Let $d \geq 1$ and let $(\Delta^{(0)}, \dots \Delta^{(d-1)}) \in (\mathbb{F}_2^n)^d$ be an independent family and $V := \left\langle \Delta^{(0)}, \dots, \Delta^{(d-1)} \right\rangle$. Then:*

$$\forall x \in \mathbb{F}_2^n, \quad D_{\Delta^{(d-1)}} \dots D_{\Delta^{(0)}} F(x) = \sum_{v \in V} F(x+v). \tag{3.1}$$

Proof. We prove it by induction on d. If $d = 1$, then Eq. (3.1) holds as it coincides with Definition 2.23. Let us now assume that Eq. (3.1) holds for $d \geq 1$. Let $(\Delta^{(0)}, \dots, \Delta^{(d)})$ be an independent family. Let $W := \left\langle \Delta^{(0)}, \dots, \Delta^{(d-1)} \right\rangle$ and $V := \left\langle \Delta^{(0)}, \dots, \Delta^{(d)} \right\rangle$. Let $x \in \mathbb{F}_2^n$. Then:

$$\begin{aligned} D_{\Delta^{(d)}} \dots D_{\Delta^{(0)}} F(x) &= D_{\Delta^{(d)}} \left(D_{\Delta^{(d-1)}} \dots D_{\Delta^{(0)}} F \right)(x) \\ &= D_{\Delta^{(d-1)}} \dots D_{\Delta^{(0)}} F\left(x + \Delta^{(d)}\right) + D_{\Delta^{(d-1)}} \dots D_{\Delta^{(0)}} F(x) \\ &= \sum_{w \in W} F\left(x + w + \Delta^{(d)}\right) + \sum_{w \in W} F(x+w) \\ &= \sum_{v \in V} F(x+v), \end{aligned}$$

where we used the induction hypothesis for the third equality. Lemma 3.1 therefore holds for any $d \geq 1$. □

It is therefore natural to define the derivation along a linear subspace.

Definition 3.2 (Higher-order derivative [Lai94])**.** Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$. Let V be a linear subspace of $\mathbb{F}_2^n$. The *derivative of F along V* (or *with respect to V*) is the function $D_V F\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$ defined by:

$$\forall x \in \mathbb{F}_2^n, \quad D_V F(x) := \sum_{v \in V} F(x+v).$$

It corresponds to the successive derivations along $\Delta^{(0)}, \dots, \Delta^{(d-1)}$, for any basis $(\Delta^{(0)}, \dots, \Delta^{(d-1)})$ of V. The dimension of V is called the *order* of the derivative D_V. ▷

Remark 3.3. With Definition 3.2, we restrict ourselves to successive derivations in *independent* directions. This is explained by the fact that deriving in colinear directions leads to the zero function. Indeed, let V be a subspace and $\Delta \in V$. Then:

$$\begin{aligned} D_\Delta D_V F(x) &= D_V F(x + \Delta) + D_V F(x) \\ &= \sum_{v \in V} F(x+v) + \sum_{v \in V} F(x + \Delta + v) \\ &= \sum_{v \in V} F(x+v) + \sum_{v \in V} F(x+v) = 0, \end{aligned}$$

where we use the $v \leftarrow v + \Delta$ as change of variables to obtain the third equality. ▷

From Definition 3.2, we observe that the derivative of a function F along V can be evaluated at any point, at the cost of $2^{\dim(V)}$ evaluations of F at *chosen* points. Despite the cost that is growing exponentially with the dimension of V, this implies that some properties of its derivatives can be used to distinguish F from a random function or permutation. We therefore list the main interesting properties about higher-order derivatives.

First, as we can expect with the intuition due to polynomial derivatives, the more we derive, the more the degree decreases.

Proposition 3.4 (Derivation and degree fall [Lai94, Proposition 2])**.** *Let* $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$*. Let* V *be a subspace. Then* $\deg_a(D_V(F)) \leq \max\left(0, \deg_a(F) - \dim(V)\right)$.

Proof. We only prove the fact for any $\Delta \in \mathbb{F}_2^n$, $\deg_{\mathrm{a}}(D_\Delta(F)) \leq \deg_{\mathrm{a}}(F) - 1$, as the original statement can be directly deduced by induction on the dimension of V. Furthermore, because the algebraic degree of F is the maximum over the algebraic degree of its coordinates, it is sufficient to prove it for a Boolean function $f\colon \mathbb{F}_2^n \to \mathbb{F}_2$, and, because it is clear from Definition 2.23 that $D_\Delta(f+g) = D_\Delta f + D_\Delta g$, it is sufficient to prove it for $f\colon (x_0, \ldots, x_{n-1}) \mapsto x^u$ for some $u \in \mathbb{F}_2^n$. Let $\Delta \in \mathbb{F}_2^n$. Then for any $x \in \mathbb{F}_2^n$:

$$D_\Delta f(x) = x^u + (x+\Delta)^u = x^u + \sum_{v \preceq u} \Delta^{u+v} x^v = \sum_{v \preceq u, v \neq u} \Delta^{u+v} x^v .$$

Therefore $\deg_{\mathrm{a}}(D_\Delta f) \leq \mathrm{wt}(u) - 1 = \deg_{\mathrm{a}}(f) - 1$. □

Furthermore, the close link between the definition of $D_V F$ and F enables us to derive the ANF of $D_V F$ from the one of F, when V is *aligned* with the canonical basis. Recall from Eq. (2.2) that we denote by $\xi^{(i)}$ the i-th vector of the canonical basis of $\mathbb{F}_2^n$. For any $u \in \mathbb{F}_2^n$, we also denote by $\mathrm{Prec}(u)$ and $\mathrm{Succ}(u)$ the sets defined by:

$$\mathrm{Prec}(u) := \left\{x \in \mathbb{F}_2^n, x \preceq u\right\}, \quad \mathrm{Succ}(u) := \left\{x \in \mathbb{F}_2^n, u \preceq x\right\}.$$

While $\mathrm{Prec}(u)$ is a linear space of dimension $\mathrm{wt}(u)$, $\mathrm{Succ}(u)$ is an affine space of dimension $n - \mathrm{wt}(u)$.

Proposition 3.5 (ANF of $D_V f$ where $V = \mathrm{Prec}(u)$ [DS09, Theorem 1])**.** *Let* $f\colon \mathbb{F}_2^n \to \mathbb{F}_2$ *be defined by* $x \mapsto \sum_{v \in \mathbb{F}_2^n} a_v x^v$*. Let* $u \in \mathbb{F}_2^n$ *and* $V = \mathrm{Prec}(u)$*. Then* f *can be decomposed as:*

$$f = D_V f \cdot X^u + g,$$

where g *contains no monomial* X^v*, such that* X^u *divides* X^v*. Equivalently,* $D_V F$ *can be expressed as:*

$$D_V f := \sum_{w \in \mathrm{Succ}(u)} a_w X^{w+u}.$$

Proof. The equivalence between the two statements is immediate, so we only prove the second one. Furthermore, we restrict ourselves to the case where f is defined by $(x_0, \ldots, x_{n-1}) \mapsto x^a$ for some $a \in \mathbb{F}_2^n$, as the general case is deduced from this particular one by invoking the linearity of the operator D_V. In that case, the equality that must be proved is the following:

$$\forall x \in \mathbb{F}_2^n, \quad D_V f(x) = \begin{cases} x^{a+u} & \text{if } u \preceq a, \\ 0 & \text{otherwise.} \end{cases}$$

Let $x \in \mathbb{F}_2^n$. First, we observe that:

$$\begin{aligned} D_V f(x) &= \sum_{v \in V} (x+v)^a \\ &= \sum_{v \preceq u} \prod_{i \in \mathrm{Supp}(a)} (x_i + v_i) \\ &= \sum_{v \preceq u} \prod_{i \in \mathrm{Supp}(a) \setminus \mathrm{Supp}(u)} x_i \prod_{i \in \mathrm{Supp}(u) \cap \mathrm{Supp}(a)} (x_i + v_i), \end{aligned}$$

where the last equality comes from the fact that for any $i \notin \mathrm{Supp}(u)$, and any $v \preceq u$ we necessarily have $v_i = 0$. Therefore, by factoring by $\prod_{i \in \mathrm{Supp}(a) \setminus \mathrm{Supp}(u)} x_i$, we obtain:

$$D_V f(x) = \left(\prod_{i \in \mathrm{Supp}(a) \setminus \mathrm{Supp}(u)} x_i \right) \left(\sum_{v \preceq u} \prod_{i \in \mathrm{Supp}(u) \cap \mathrm{Supp}(a)} (x_i + v_i) \right).$$

Let $b \in \mathbb{F}_2^n$ be such that $\mathrm{Supp}(b) = \mathrm{Supp}(u) \cap \mathrm{Supp}(a)$. Let $g \colon x \mapsto x^b$. We can rewrite the previous equality as:

$$D_V f(x) = \left(\prod_{i \in \mathrm{Supp}(a) \setminus \mathrm{Supp}(u)} x_i \right) D_V g(x).$$

But $D_V g$ is a derivative of order $\mathrm{wt}(u)$ of a function of algebraic degree $\mathrm{wt}(b)$, with $\mathrm{wt}(b) \leq \mathrm{wt}(u)$. So, by Proposition 3.4, this implies that $D_V g$ is a constant function whose value is $D_V g(0)$. By observing that $\forall v, w \in \mathbb{F}_2^n$, $v^w = 1$ if and only if $w \preceq v$, we obtain:

$$D_V g(0) = \sum_{v \preceq u} v^b = \sum_{b \preceq v \preceq u} 1.$$

If $u \preceq a$, then $b = u$ and the only term in the previous sum is the one for $v = u$, so $D_V g(0) = 1$. We conclude in that case that $D_V f(x) = x^{u+a}$ by observing that, $\prod_{i \in \mathrm{Supp}(a) \setminus \mathrm{Supp}(u)} x_i = x^{u+a}$. Otherwise, $u \not\preceq a$, so $b \preceq u$, with $b \neq u$. This implies that the sum is of even size $2^{\mathrm{wt}(u) - \mathrm{wt}(b)}$, and therefore $D_V g(0) = 0$. □

Because of Proposition 3.5, higher-order derivatives along $\mathrm{Prec}(u)$ have attracted more attention. Note that those linear spaces are exactly the linear spaces spanned by a subset of the canonical basis. In this sense, those spaces are *aligned* with the canonical basis. Recently, Hu, Peyrin, Tan & Yap [Hu+23] employed new tools to study properties of derivatives along non-aligned spaces by introducing auxiliary functions with twice as many variables as the original function $f \colon \mathbb{F}_2^n \to \mathbb{F}_2$. We stress here that this is not necessary. After all, a space $V := \left\langle \Delta^{(0)}, \ldots, \Delta^{(d-1)} \right\rangle$ is aligned with respect to any basis that contains the vectors $\Delta^{(0)}, \ldots, \Delta^{(d-1)}$.

Proposition 3.6 (Realignment of non-aligned derivatives)**.** *Let $f\colon \mathbb{F}_2^n \to \mathbb{F}_2$. Let $\Delta^{(0)}, \ldots, \Delta^{(d-1)} \in (\mathbb{F}_2^n)^d$ be an independent family. Let $A\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ be a bijective linear mapping satisfying $A(\xi^{(i)}) = \Delta^{(i)}$ for any $i \in [\![0, d-1]\!]$. Let $V := \left\langle \Delta^{(0)}, \ldots, \Delta^{(d-1)} \right\rangle$ and $W := \left\langle \xi^{(0)}, \ldots, \xi^{(d-1)} \right\rangle$ Then:*

$$D_V f = D_W (f \circ A) \circ A^{-1}.$$

Proof. By a direct computation, we observe that:

$$D_V f \circ A = \sum_{v \in V} f(A(X) + v) = \sum_{w \in W} f(A(X) + Aw) = \sum_{w \in W} f \circ A(X + w) = D_W(f \circ A),$$

where we use the fact that A maps bijectively W onto V, and the linearity of A. Finally, because A is bijective, we further obtain $D_V f = D_W(f \circ A) \circ A^{-1}$. □

Proposition 3.6 therefore states that the derivative along any space can be obtained from the ANF of a linear-equivalent function.

Example 3.7 (Non-aligned derivative)**.** Let $f\colon \mathbb{F}_2^3 \to \mathbb{F}_2$ be the function introduced in [Hu+23, Example 1], *i.e.* f is defined by:

$$f\colon (x_0, x_1, x_2) \mapsto x_0 x_1 x_2 + x_0 x_1 + x_0 x_2 + x_1 x_2.$$

Let us compute the same derivative as the one considered in this paper, that is, $D_V f$ where $V = \left\langle \Delta^{(0)}, \Delta^{(1)} \right\rangle$ with $\Delta^{(0)} = (1, 0, 1)$ and $\Delta^{(1)} = (1, 1, 1)$. Let us consider the following matrices:

$$A := \begin{array}{c} \begin{array}{ccc} \Delta^{(0)} & \Delta^{(1)} & \xi^{(2)} \end{array} \\ \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 1 \end{pmatrix} \end{array} \quad \text{and} \quad A^{-1} := \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix}.$$

We first compute $f \circ A$. For any $x_0, x_1, x_2 \in \mathbb{F}_2$, we get:

$$\begin{aligned} f \circ A(x_0, x_1, x_2) &= f(x_0 + x_1, x_1, x_0 + x_1 + x_2) \\ &= (x_0 + x_1) x_1 (x_0 + x_1 + x_2) + (x_0 + x_1) x_1 + \\ &\quad (x_0 + x_1)(x_0 + x_1 + x_2) + x_1 (x_0 + x_1 + x_2) \\ &= x_0 x_1 x_2 + x_0 x_1 + x_0 x_2 + x_1 x_2 + x_0 \\ &= (x_2 + 1) x_0 x_1 + (x_0 x_2 + x_1 x_2 + x_0). \end{aligned}$$

Let $W = \left\langle \xi^{(0)}, \xi^{(1)} \right\rangle$. By applying Proposition 3.5, we get:

$$\forall (x_0, x_1, x_2) \in \mathbb{F}_2^3, \quad D_W(f \circ A)(x_0, x_1, x_2) = x_2 + 1.$$

Finally, we observe that $D_W(f \circ A) \circ A^{-1}(x_0, x_1, x_2) = x_0 + x_2 + 1$. Therefore, as announced in [Hu+23, Example 1], we obtain $D_V f : (x_0, x_1, x_2) \mapsto x_0 + x_2 + 1$, but in a much more direct way. ▷

3.1.2 Cryptanalysis based on higher-order derivatives

Many cryptanalysis techniques leverage properties of higher-order derivatives, one way or another. These techniques are included in the class of *integral attacks* [KW02], because of the summation process that appears in Definition 3.2. In the following, we make distinction between the attacks that are distinguishers, or based on distinguishers, from the ones that are key-recovery attacks by nature.

3.1.2.a Higher-order differential distinguishers

The first higher-order differential cryptanalysis dates back to a paper of Knudsen [Knu95] in which the author presents a higher-order distinguishing property on Feistel networks using quadratic Sboxes (over $\mathbb{F}_p$ with odd p). This attack relies on the fact that any second-order derivative of a quadratic function is necessarily constant. This distinguisher is then used to mount a last-round attack.

The fact that a derivative of a cryptographic function might be constant is the most prominent distinguishing property based on higher-order derivatives. In the case of a block cipher $\mathcal{E} = (E_k \colon \mathbb{F}_2^n \xrightarrow{\sim} \mathbb{F}_2^n)_{k \in \mathbb{F}_2^\kappa}$, this amounts to proving that there exists a subspace V such that for any k, $D_V E_k$ is not only constant, but also that the constant value $D_V E_k(0)$ is the same for all k, *i.e.* independent of k.

Soon after the work by Knudsen, the so-called[1] *Square attack* [DKR97] or *saturation attacks* [Luc02] tackled this problem. These attacks are structural in nature: they do not rely on the actual choice of Sbox, but only on its bijectivity, and they often focus on diffusion at the byte (or 4-bit nibble) level, rather than the bit level. By analyzing diffusion of the input bits due to the *overall* structure of the cipher, the authors of these works exhibit such constant derivatives.

The Square attack. Let us describe in more detail the 3-round Square attack on the AES, that is depicted in Figure 3.1. Let $k \in \mathbb{F}_2^\kappa$. This attack considers

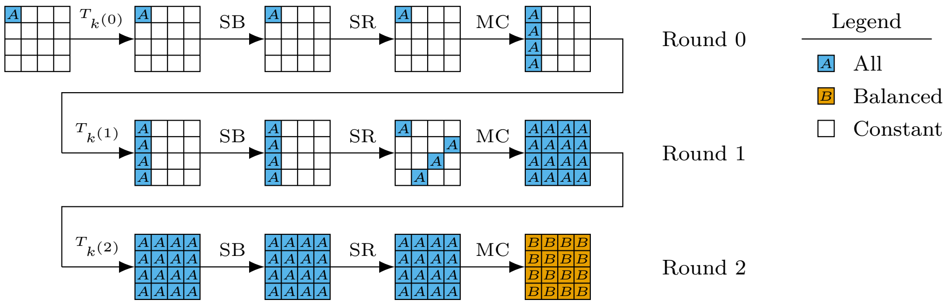


Figure 3.1: Three-round Square attack on AES-like ciphers. *All* stands for "all values are taken", *Balanced* for "the values sum to 0", and *Constant* for "all values are identical".

[1] The name Square attack refers to the "dedicated attack" mounted against the block cipher Square [DKR97, Section 6].

$D_V E_k = \sum_{v \in V} E_k(X + v)$ where $V = (\mathbb{F}_2)^8 \times \{0\} \ldots \times \{0\}$. To do so, it follows the evolution of the affine spaces $a + V$ for any a through the cipher. Because the constant addition and the Sbox are parallel applications of bytewise bijections, we know that the image of an affine coset of V by $\mathcal{S} \circ T_{k(0)}$ is still a coset of V. The ShiftRows operation only reorganizes the bytes. Regarding the MixColumns function, it applies the linear function $M \colon (\mathbb{F}_{2^8})^4 \to (\mathbb{F}_{2^8})^4$ that is defined by:

$$M \colon x \mapsto (M_0(x), M_1(x), M_2(x), M_3(x)),$$

where for any $i \in [\![0, 3]\!]$, M_i is defined by:

$$M_i(x_0, x_1, x_2, x_3) \mapsto a_{i,0}x_0 + a_{i,1}x_1 + a_{i,2}x_2 + a_{i,3}x_3,$$

with $a_{i,j} \neq 0$ for all i, j. If the set in input is $\{a\} \times \{b\} \times \{c\} \times \{d\}$, with $a, b, c, d \in \mathbb{F}_2^8$, the image is again a one-point set. But if the set is $\mathbb{F}_2^8 \times \{b\} \times \{c\} \times \{d\}$ each output byte takes each possible value. Indeed, as $a_{0,0} \neq 0$, the function $x_0 \mapsto a_{0,0}x_0 + a_{0,1}b + a_{0,2}c + a_{0,3}c$ is a bijective affine mapping, and the same holds for the other coordinates because of the natural symmetries. These rules enable us to follow the evolution until the third MixColumns operation. This time, the input set of each column takes all values on each byte. Let us denote by $x^{(j)}$ for $j \in [\![0, 255]\!]$ the 256 input values. Let us denote by $F_k \colon \mathbb{F}_2^{128} \to \mathbb{F}_2^8$ the function which corresponds to the the first output byte after three rounds of AES. By summing over all output values, we obtain:

$$\begin{aligned} D_V F_k(0) = \sum_{j=0}^{255} M_0(x_0^{(j)}, x_1^{(j)}, x_2^{(j)}, x_3^{(j)}) &= \sum_{j=0}^{255} a_{0,0}x_0^{(j)} + a_{0,1}x_1^{(j)} + a_{0,2}x_2^{(j)} + a_{0,3}x_3^{(j)} \\ = a_{0,0} \sum_{j=0}^{255} x_0^{(j)} + a_{0,1} \sum_{j=0}^{255} x_1^{(j)} &+ a_{0,2} \sum_{j=0}^{255} x_2^{(j)} + a_{0,3} \sum_{j=0}^{255} x_3^{(j)}. \end{aligned}$$

But as $\sum_{j=0}^{255} x_0^{(j)} = \sum_{x \in \mathbb{F}_2^8} x = 0$, we conclude that $D_V F_k(0) = 0$. The same can be done for any output byte, so we deduce that $D_V E_k(0) = 0$. Finally, everything remains true if the input space V is replaced by $V + a$ for some $a \in \mathbb{F}_2^{128}$, so we can conclude that $D_V E_k$ is the zero function. This serves as a distinguisher as the analysis is independent of the key k. It is furthermore independent of the actual Sbox used and still holds if distinct bijective Sboxes are used at each round and at each byte position.

Note that the best key-recovery attacks against 6-round AES are based on enhancements of this distinguisher [Fer+01, Dun+24]. We refer to the thesis of Bariant [Bar24, Sections 2.3.3.2 & 4.1] for an up-to-date report on the cryptanalysis of AES. Finally, in line with Square attacks, Biryukov & Shamir [BS01] use the same methodology in *multiset attacks*.

Distinguisher based on degree bounds. Another way of finding such distinguishers is by carefully analyzing the growth of degree. Because of Proposition 3.4, if an upper bound D on the degree is known, the derivative along any subspace V of dimension $\dim(V) \geq D$ is constant. Moreover, if $\dim(V) \geq D+1$, then the derivative is the zero function. The distinguishing attack then consists in evaluating the derivative at one point, at the cost of $2^{\dim(V)}$ chosen plaintexts, and verifying whether or not its value is 0. If $D_V F$ is a function with m output bits, this is expected at random with probability $\frac{1}{2^m}$. This is the reason why bounds on the degree of iterated constructions often come with associated distinguishers on cryptographic primitives. Examples are given in Section 2.3.1.c.

Among all choices of directions V, spaces of the form $\operatorname{Prec}(u)$ for some $u \in \mathbb{F}_2^n$ play a particular role, and among all points where the derivative along V can be evaluated at, 0 plays also a particular role. Indeed, let $f\colon \mathbb{F}_2^n \to \mathbb{F}_2, x \mapsto \sum_{v \in \mathbb{F}_2^n} a_v x^v$. In that case, if $V = \operatorname{Prec}(u)$ for some u, the value of $D_V f(0)$ is expressed as:

$$D_V f(0) = \sum_{v \in V} f(v) = \sum_{x \preceq u} f(v) = a_v, \tag{3.2}$$

where we used Proposition 2.10 to obtain the last equality. The evaluation of a derivative in that case corresponds to an ANF coefficient. Therefore, ensuring that a specific coefficient does not appear in any coordinate of a cryptographic function $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$ is another way of finding a higher-order differential distinguisher.

Division property. In 2015, Todo [Tod15b] tackled this problem by introducing the division property which generalizes the fact that a set of values sums to 0.

Definition 3.8 (Division property [Tod15b, BC16])**.** A set $Z \subset \mathbb{F}_2^n$ is said to fulfill the division property of order $k \in [\![1, n]\!]$ if it satisfies:

$$\forall u \in \mathbb{F}_2^n, \operatorname{wt}(u) < k, \quad \sum_{x \in Z} x^u = 0.$$

Equivalently, Z fulfills the division property of order k if its indicator function $\mathbf{1}_Z$ is of degree $n-k$, where $\mathbf{1}_Z$ is defined by:

$$\mathbf{1}_Z\colon \mathbb{F}_2^n \to \mathbb{F}_2, x \mapsto \begin{cases} 1 & \text{if } x \in Z, \\ 0 & \text{otherwise.} \end{cases}$$

▷

The equivalent definition is due to Boura & Canteaut [BC16]. Thanks to it, we observe that the division property of order 2 is equivalent to having a set Z of even cardinality whose elements sum to 0. From there, Todo studies the evolution of the division property of subspaces of the form $a + \operatorname{Prec}(u)$ through the round function. In the end, he finds values for $u \in \mathbb{F}_2^n$ such that $E_k(a + \operatorname{Prec}(u))$ satisfies the division property of order 2 for any $a \in \mathbb{F}_2^n$. This exactly corresponds to the fact that $D_V E_k = 0$ where $V = \operatorname{Prec}(u)$. Contrary to the Square attack, the method of Todo takes into account the degree of the Sbox S, while the technique of Boura

& Canteaut leverages the ANF of S in more detail. This technique happens to be extremely powerful as highlighted by the first attack on the full block cipher Misty1 [Tod15a, Tod17].

Exact division property. The division property is however heuristic and does not guarantee that a distinguisher will be found, even if it exists. This is the reason why a line of papers [Tod15b, Tod17, TM16, Hao+20, Hao+21] approached the problem of refining this method with the help of automated solvers. After a few years, the exact formalisms [BC16, Hu+20, BV23] that were suggested all sum up to being able to compute exactly (part of) the ANF of a cryptographic primitive that is iteratively built.

This iterated construction naturally leads to notions of trails, such as *division trails* [Xia+16], *monomials trails* [Heb+20, Hu+20], or *algebraic trails* [BV23], which are all closely related. The link between *parity sets* [BC16], division trails and monomial trails is clearly established by Hebborn, Leander & Udovenko [HLU23, Section 3.2]. Beyne & Verbauwhede [BV23, Section 4.1] complete this classification with the link with algebraic trails. This latest study [BV23] also introduces a matrix point-of-view, similiar to the ones regarding differential and linear cryptanalysis that are presented in Sections 2.3.3.d and 2.3.4.c.

As times goes, the name *division property* has been overloaded. It now often refers to any algorithmic method to recover part of the ANF of an iterated construction, and not the Definition 3.8 anymore.

3.1.2.b Higher-order differential key-recoveries

On the other side of the scope of higher-order differential attacks are the attacks that are key-recovery *by design*. Those attacks are often called *cube attacks* due to the work of Dinur & Shamir [DS09]. Yet their infancy dates back to the AIDA attack by Vielhaber [Vie07]. The idea of such attacks is to obtain equations in key bits in order to gain partial or full knowledge about the key thanks to the solving of a system. But contrary to the standard scenario described in Section 2.3.1.b, the equations are not derived by using $y = E_k(x)$ for a single known plaintext/ciphertext pair (x, y). Instead, equations of the form $y = D_V E_k(x)$ are derived using $2^{\dim(V)}$ chosen plaintexts.

More precisely, let $\mathcal{E} = (E_k \colon \mathbb{F}_2^n \xrightarrow{\sim} \mathbb{F}_2^n)_{k \in \mathbb{F}_2^\kappa}$. Let $V \subset \mathbb{F}_2^n$, and $x \in \mathbb{F}_2^n$. An attacker that is interested in the family $(D_V E_k(x))_{k \in \mathbb{F}_2^\kappa}$, can consider the function $F^{V,x}$ defined by:

$$F^{V,x} \colon \mathbb{F}_2^\kappa \to \mathbb{F}_2^n, k \mapsto D_V E_k(x),$$

and look at the ANF of each of its coordinates:

$$\forall i \in [\![0, n-1]\!], \quad F_i^{V,x} \colon k = (k_0, \ldots, k_{\kappa-1}) \mapsto \sum_{v \in \mathbb{F}_2^\kappa} b_{v,i} k^v.$$

From there, with a blackbox access to the instantiated cipher E_k for an unknown key $k \in \mathbb{F}_2^\kappa$, the adversary can query the encryption of all plaintexts in $x + V$, and recover the value of $y := D_V E_k(x) = \sum_{v \in V} E_k(x+v)$. Therefore, the following equations in unknowns $k_0, \ldots, k_{\kappa-1}$ are obtained:

$$\forall\ i \in [\![0, n-1]\!]\,, \quad y_i = \sum_{v \in \mathbb{F}_2^\kappa} b_{v,i} k^v .$$

Such a system can finally be solved as explained in Section 2.3.1.b. Traditionally, such cube attacks are presented in two steps: the offline phase during which the attacker recovers the ANF of $F^{V,x}$ and the online one, where he asks for encryption and solves the system deduced from the queries. Two major challenges are therefore brought to an adversary.

1. First, while the ANF of $F^{V,x}$ is theoretically at hand (because of Kerckhoff's principle), it is necessary for its recovery to be *effective.*

2. Secondly, once the system of equations has been mounted, it should also be *solvable at reasonable cost.*

Expression of key-dependent coefficients. To cope with the first challenge, the first simplification that is usually (if not, always) made is to only consider pairs (V, x) of the form $(\mathrm{Prec}(u), 0)$ for some $u \in \mathbb{F}_2^n$. Such space $\mathrm{Prec}(u)$ can be considered as a *cube* in $\mathbb{F}_2^n$, hence the name of the attack.

As shown by Eq. (3.2), this simplification amounts to looking for the expression of a key-dependent coefficient in the ANF of E_k. More precisely, it is possible to look at $\mathcal{E}$ as a function in both key and plaintext variables:

$$\mathcal{E} \colon \mathbb{F}_2^n \times \mathbb{F}_2^\kappa \to \mathbb{F}_2^n .$$

In that case, for any $i \in [\![0, n-1]\!]$ the i-th coordinate can be expressed as:

$$\mathcal{E}_i \colon (x_0, \ldots, x_{n-1}, k_0, \ldots, k_{\kappa-1}) \to \sum_{u \in \mathbb{F}_2^n} a_{u,i} x^u ,$$

where each $a_{u,i}$ is a function of k: $a_{u,i} := \sum_{v \in \mathbb{F}_2^\kappa} b_{u,i,v} k^v$.

Deriving along $V = \mathrm{Prec}(u)$ then corresponds to deriving along $W = \mathrm{Prec}((u,0)) \subset \mathbb{F}_2^n \times \mathbb{F}_2^\kappa$. Indeed, for any $k \in \mathbb{F}_2^\kappa$, we obtain:

$$\begin{aligned} D_V E_{k,i}(0) &= \sum_{v \in V} E_{k,i}(v) = \sum_{v \in V} \mathcal{E}_i(v,k) \\ &= \sum_{w \in W} \mathcal{E}_i\left((v,k) + w\right) = D_W \mathcal{E}_i(0,k) = a_{u,i}(k). \end{aligned}$$

In that case, this implies that the function $F^{V,0}$, that is a function of k, coincides with the function $a_{u,i}$. So, according to Proposition 3.5, recovering its ANF can be done by recovering part of the ANF of $\mathcal{E}_i$. This can for instance be done using the algorithms mentioned in Section 3.1.2.a. Before the exact formalisms were established, some heuristic methods were used to probe the ANF and search for simple key-dependent coefficients. Some of them were later proved to actually be key-independent, and only usable as distinguishers [YT19]. The exact methods were also not developed at the time of [DS09]. Instead, Dinur & Shamir target coefficients $a_{u,i}$ which are linear in key variables. To do so, because a full access to the cipher is given during the offline phase, they can compute $a_{u,i}(k)$ for many keys k, k'. If $a_{u,i}(k) + a_{u,i}(k') = a_{u,i}(k + k')$, for many keys, they consider $a_{u,i}$ to be linear and interpolate it. This solves both of the problems that are presented above as interpolating a linear polynomial is not too costly, and solving the obtained linear equations is simple.

Finding effective cubes. In the general case, the second challenge is the hardest of both. While it seems at first sight to be a challenge of the online phase, it is not the case. Indeed, the effective solving of the system is entirely dependent on the simplicity of the equations. It is therefore necessary to target, during the offline phase, functions $a_{u,i} \colon \mathbb{F}_2^\kappa \to \mathbb{F}_2$ as simple as possible, for instance, of low degree and/or in few variables and/or as sparse as possible. This is actually the hardest task to mount a practical cube attack.

The main heuristic to cope with this problem is based on bounds on the algebraic degree of $\mathcal{E} \colon \mathbb{F}_2^n \times \mathbb{F}_2^\kappa \to \mathbb{F}_2^n$. If such a bound D is known, the coefficient $a_{u,i}$ associated to a monomial of degree $\mathrm{wt}(u)$ (or equivalently to a cube of dimension $\mathrm{wt}(u)$) is necessarily a polynomial in variables $k_0, \ldots, k_{\kappa-1}$ of degree $\deg_{\mathrm{a}}(a_{u,i}) \leq D - \mathrm{wt}(u)$. This heuristic is already used by Dinur & Shamir in their seminal work [DS09]: in order to look for equations of low degree in k, they target cubes of high degree in x. This however comes with a strong caveat as the data complexity is exponential in the dimension of the cube. Building an effective cube attack sums up to finding a good trade-off between data complexity and time complexity to solve the system. In this context, this trade-off is a balance between the degree of x^u that should not be too high, and the expression of the corresponding $a_{u,i}$ that should not be too intricate.

In conditional cube attacks [Hua+17], a different notion of simplicity for the coefficients $a_{u,i}$ is considered: the family $(a_{u,i})_{i \in \mathbb{F}_2^n}$ is considered simple enough if it has a simple common divisor. Such property is exploited in previous works that are presented in Section 3.3, but also in our work described in Section 3.4.

Vocabulary. The term cube attack is in practice ill-defined. As for higher-differential attacks, different names can be given to the same notion. For instance, the function $x, k \mapsto D_V E_k(x)$ or when x is fixed, the function $k \mapsto D_V E_k(x)$ are sometimes known as *superpoly* associated to V. The derivative along a cube $\mathrm{Prec}(u)$ is sometimes known as the *cube-sum*. Furthermore, as the division property before it, the name cube attack is overloaded. It often refers to key-recovery as just described, but *cube distinguishers* or *cube tester* [Aum+09] refer to higher-order differential distinguishers as described in Section 3.1.2.a. We continue using the latter naming.

With this formalism in mind, we give concrete applications of higher-order differential attacks against (round-reduced) Ascon. Section 3.2 is dedicated to the precise description of Ascon, while Section 3.3 reviews the previous analyzes of Ascon, with respect to higher-order differential attacks. The practical cube attack against nonce-misused Ascon, that was mounted collaboratively with Anne Canteaut & Léo Perrin, is presented in Section 3.4.

3.2 Description of Ascon

3.2.1 The Ascon family

As already mentioned, Ascon [Dob+21] is a family of symmetric primitives designed for lightweight use cases. In particular, part of the family was selected in the final portfolio of the CAESAR [Cae13] competition, in the "lightweight applications" category. Ascon was also selected by the NIST in February 2023 as the winner of the lightweight standardization process. It is currently being standardized. This puts Ascon in the spotlight of lightweight cryptography, and demands a continued cryptanalysis effort of this family.

The Ascon suite is made of:

- three authenticated encryptions with associated data (AEAD [Rog02]), namely Ascon-128, Ascon-128a & Ascon-80pq;
- two hash functions, namely Ascon-Hash, and Ascon-Hasha;
- and two extendable-output functions, namely Ascon-Xof, and Ascon-Xofa.

Even if we focus more on Ascon-128 in the remaining of this chapter, it is important to note that all members of the Ascon family share their two main components:

- both the AEAD mode and the hashing mode are derived from the sponge construction [Ber+07], and
- these modes are instantiated using the same cryptographic permutation $p\colon \mathbb{F}_2^{320} \xrightarrow{\sim} \mathbb{F}_2^{320}$, which is iterated a possibly-different number of times.

It is therefore natural that all members also share the same state size of 320 bits, and the same security claim of 128 bits, even though the key size of Ascon-80pq is 160 bits.

In the following, we first describe the encryption mode shared by the three AEAD ciphers. The cryptographic permutation $p\colon \mathbb{F}_2^{320} \to \mathbb{F}_2^{320}$ is described in Section 3.2.3. Note that the usage of p is in conflict with the principle of using uppercase for vectorial Boolean functions that is introduced in Section 2.2. Nonetheless, we continue using this notation introduced by the designers in the remaining of this chapter. Furthermore, *if not stated otherwise*, Ascon now refers to Ascon-128.

3.2.2 The AEAD mode of Ascon

As any AEAD primitive, Ascon-128 aims to provide integrity, confidentiality, together with authenticity in an effective integrated manner [BN00]. It must also ensure the authenticity of associated public data.

To do so, the design of the AEAD versions are based on the well-studied Sponge Duplex mode of operation [Ber+07, Ber+12a], but with initialization and finalization strengthened by a feed-forward addition of the key.

This mode is depicted in Figure C.1, where $\mathrm{IV} \in \mathbb{F}_2^{64}, N \in \mathbb{F}_2^{128}, T \in \mathbb{F}_2^{128}$ respectively stands for initial value, nonce, and tag. Furthermore, the i-th 64-bit blocks of associated data, plaintext and ciphertext are denoted by $A^{(i)}, P^{(i)}, C^{(i)}$ respectively. Finally, s_r and s_c are respectivelly called *rate* and *capacity*. They are defined as the sizes of the *outer state* and *inner state*. The outer state corresponds to the part of the state in which external values are added, and from which ciphertext blocks are output. On the other hand, the inner state is the part of the state that, ideally at least, is not available to an adversary. For Ascon-128, the rate is $s_r = 64$ bits and the capacity is $s_c = 256$. We also denote by $\Sigma_{\mathrm{AD}}, \Sigma_{\mathrm{E}}, \Sigma_{\mathrm{F}}$ the 320-bit state before the processing of associated data, before encryption and before finalization. The notation $||$ stands for the concatenation of binary words. The numbers r_{out} and r_{in} are the amount of time the permutation p is iterated, during initialization and finalization for r_{out}, and during encryption for r_{in}. For Ascon-128, r_{out} is equal to 12 and r_{in} to 6.

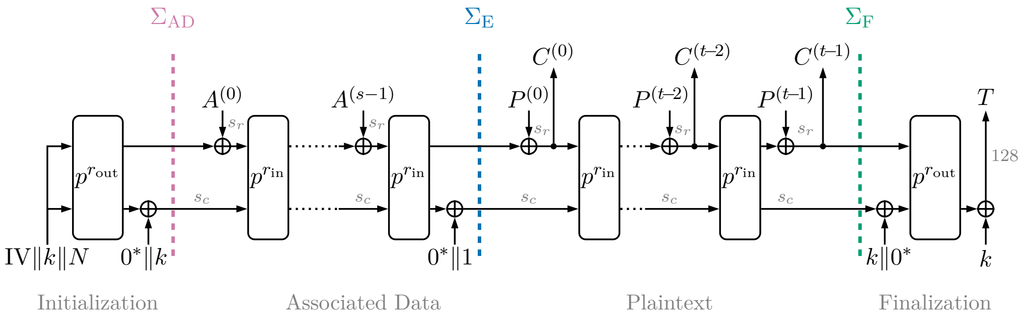


Figure 3.2: The AEAD encryption mode of Ascon.

Ascon-128 is then an instantiation of this mode, with the already-introduced values for the parameters $r_{\mathrm{out}}, r_{\mathrm{in}}, s_c, s_r$, and with the round permutation p that is described in the following section. The security level claimed for Ascon-128 is 128 bits in terms of plaintext confidentiality, and plaintext/data/nonce integrity. However, these claims are made under three hypothesis:

1. The *single* usage of each nonce,
2. the encryption of less than 2^{64} blocks for each key, and

3. the output of a decrypted plaintext only if the tag is correct.

3.2.3 The round function p

The advantage of permutation-based cryptography is that both the design and the analysis are mainly focused on the used cryptographic permutation. In the case of Ascon, the permutation p is a 320-bit permutation, which is built by alternatively looking at $\mathbb{F}_2^{320}$ as $(\mathbb{F}_2^{64})^5$ and as $(\mathbb{F}_2^5)^{64}$. More precisely, its construction follows the logic of an SPN (see Figure 1.4): p is built as the composition of three bijective layers. We denote by $p_L, p_S, p_C \colon \mathbb{F}_2^{320} \to \mathbb{F}_2^{320}$ respectively, the linear layer, Sbox layer and constant addition. With this notation, p is defined by $p := p_L \circ p_S \circ p_C$. In order to describe the different layers, it is convenient to look at a state $X \in \mathbb{F}_2^{320}$ as a binary matrix with 5 rows and 64 columns. In that case $X^{(i)} \in \mathbb{F}_2^{64}$ denotes the i-th row of the matrix, and $X_j^{(i)}$ the j-th value of the i-th row. With this representation in mind, the outer part of the state is the first row for Ascon-128 and Ascon-80pq, and the first two rows for Ascon-128a. As described in the following paragraphs and as depicted in Figure 3.3, the round-constant addition and the linear layer are applied row-wise, while the Sbox layer is applied column-wise.

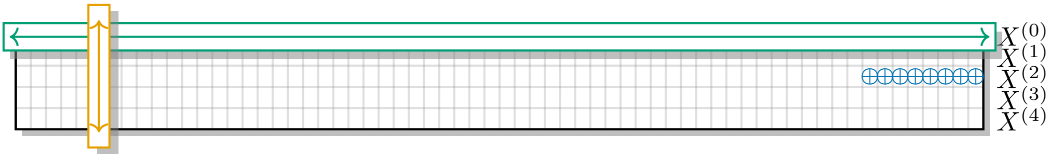


Figure 3.3: The column-wise S-box layer, the row-wise linear layer and the constant addition.

Constant addition. Despite the fact that the round index does not appear in the notation p_C, the constant addition is round-dependent. At each round, a distinct constant is added to the state. Those constants are sparse, as they affect only 8 bits of the third row $X^{(2)}$, and they are easily-computable. Indeed, from the first constant, the following ones are deduced by successively incrementing the least significant half by one, and decrementing the most significant half by one.

Substitution layer. The substitution layer of Ascon is made of 64 parallel calls to a *single* bijective 5-bit Sbox. This Sbox $S \colon \mathbb{F}_2^5 \to \mathbb{F}_2^5$ is applied to each column of the state. This choice enables the designers to easily obtain a fast bit-sliced implementation of this layer. As the outer state corresponds to the first row, it also avoids the control of all entries of a single S-box by an adversary. Regarding its properties, we remind the main ones that are already mentioned throughout Chapter 2.

- The ANF of S is given in Example 2.5. Its algebraic degree is 2.
- The DDT of S is given in Example 2.38. Its differential uniformity is $\delta_S = 8$.
- The LAT of S is given in Example 2.17. Its linearity is $\mathcal{L}(S) = 16$.
- As shown in Example 2.68, the Sbox of Ascon is affine-equivalent to the χ Sbox of the current NIST standard SHA3 [Ber+11, Nis07].

The quadraticity of S is the main property leveraged by the higher-order differentials presented in Section 3.3 and Section 3.4.

Linear layer. The linear layer of Ascon consists in 5 parallel calls to 5 distinct linear bijections $L^{(i)} \colon \mathbb{F}_2^{64} \to \mathbb{F}_2^{64}$ for $i \in [\![0, 4]\!]$. Each of them can easily be described using two additions and two cyclic shifts. For any $X^{(0)}, X^{(1)}, X^{(2)}, X^{(3)}, X^{(4)} \in \mathbb{F}_2^{64}$, they are defined by:

$$\begin{aligned}
L^{(0)}(X^{(0)}) &:= X^{(0)} \oplus \ (X^{(0)} \ggg 19) \ \oplus \ (X^{(0)} \ggg 28), \\
L^{(1)}(X^{(1)}) &:= X^{(1)} \oplus \ (X^{(1)} \ggg 61) \ \oplus \ (X^{(1)} \ggg 39), \\
L^{(2)}(X^{(2)}) &:= X^{(2)} \oplus \ (X^{(2)} \ggg 1) \ \oplus \ (X^{(2)} \ggg 6), \\
L^{(3)}(X^{(3)}) &:= X^{(3)} \oplus \ (X^{(3)} \ggg 10) \ \oplus \ (X^{(3)} \ggg 17), \\
L^{(4)}(X^{(4)}) &:= X^{(4)} \oplus \ (X^{(4)} \ggg 7) \ \oplus \ (X^{(4)} \ggg 41).
\end{aligned}$$

The iterated permutations $p^{r_{\text{out}}}$ and $p^{r_{\text{in}}}$. As shown in Figure C.1, the permutation p is iterated r_{out} times at initialization and finalization and r_{in} times during encryption. In the case of Ascon-128, r_{out} is equal to 12 and r_{in} to 6. We therefore denote by $p_L^{(i)}, p_S^{(i)}, p_C^{(i)}$ the i-th application of p_L, p_S, p_C, as depicted in Figure 3.4.

3.3 Previous higher-order differential attacks against Ascon

In this section, we sum up the higher-order differential attacks mounted by other authors. The distinguishing attacks are presented in Table 3.1 and the key-recovery attacks in Table 3.2.

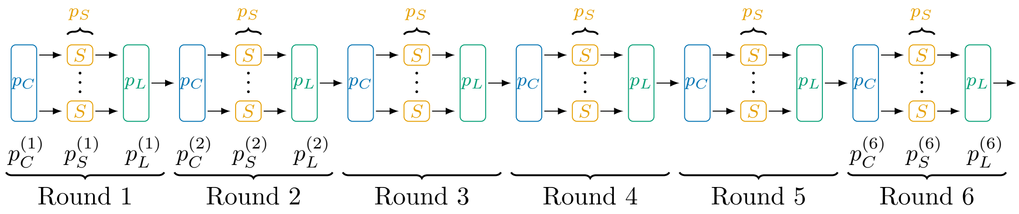

p_S
S
p_C
p_L
$p_C^{(1)}$ $p_S^{(1)}$ $p_L^{(1)}$
$p_C^{(2)}$ $p_S^{(2)}$ $p_L^{(2)}$
$p_C^{(6)}$ $p_S^{(6)}$ $p_L^{(6)}$
Round 1
Round 2
Round 3
Round 4
Round 5
Round 6

Figure 3.4: The iterated permutation p^6.

Attack type	Target	Nb of rounds	Data / Time	Method	Source
Nonce-respecting distinguisher	Permutation	12/12	2^{130}	Zero-sum	[Dob+15]
	Initialization	4/12	2^{2}	Proba. H-O	[Hu+23]
		5/12	2^{12}	Proba. H-O	[Hu+23]
		6/12	2^{33}	Deg. bound	[Dob+15]
		6/12	2^{31}	Deg. bound	[Roh+21]
		6/12	2^{17} †	Deg. bound	[RS21]
		7/12	2^{60}	Deg. bound	[Roh+21]
		7/12	2^{33} †	Deg. bound	[RS21]
Nonce-misuse distinguisher	Encryption	6/12	2^{33}	Deg. bound	[Dob+15]

† stands for weak-key subspace, *Proba. H-O* for probabilistic higher-order differential distinguisher.

Table 3.1: Summary of higher-order differential distinguishers against Ascon.

These attacks have been applied to round-reduced variants of the initialization of Ascon, but also to round-reduced versions of the encryption phase of Ascon. In that later case, key-recoveries are instead inner-state recoveries.

3.3.1 Initial cryptanalysis

The first analysis of Ascon with respect to higher-order differential attacks was made by its designers [Dob+15]. In this work, the authors mount a so-called *zero-sum distinguisher* [Aum+10, BC11] against the full permutation p^{12} with time complexity 2^{130}. This kind of distinguishers works for unkeyed permutations. First, p^{12} is decomposed as $p^{12} = p^{n_1} \circ p^{n_0}$. Then, the algebraic degrees of p^{-n_0} and p^{n_1}, that we denote by $d_0 := \deg_a(p^{-n_0})$ and $d_1 := \deg_a(p^{n_1})$, are estimated and finally a vector space V of dimension $d > \max(d_0, d_1)$ is considered. Finally, the sums over the sets $\{p^{-n_0}(v), v \in V\}$ and $\{p^{n_1}(v), v \in V\}$ are considered and must both be 0, as seen in Section 3.1.2.a.

Attack type	Target	Nb of rounds	Data / Time	Method	Source
Nonce-respecting key-recovery	Init.	5/12	$2^{19}/2^{35}$	Cube	[Dob+15]
		5/12	2^{24}	Cond. cube	[LDW17]
		5/12	$2^{22}/2^{22}$	Proba H-O.	[Hu+23]
		6/12	$2^{34}/2^{66}$	Cube	[Dob+15]
		6/12	2^{40}	Cond. cube	[LDW17]
		7/12	$2^{77.2}$ / $2^{103.9}$	Cond. cube	[LDW17]
		7/12	$2^{77.2}$ / 2^{77}	Cond. cube†	[LDW17]
		7/12	2^{64} / 2^{123}	Cube	[Roh+21]
		7/12	2^{64} / 2^{97}	Cube†	[RS21]
		7/12	2^{63} / $2^{115.2}$	Cube†	[RS21]
		7/12	2^{70} / $2^{72.4}$	Cond. cube†	[Hu24]
		7/12	2^{70} / $2^{104.7}$	Cond. cube	[Hu24]
Nonce-misuse key-recovery	Init.	7/12	? / 2^{97}	Cube-like	[Li+17]
Nonce-misuse state-recovery	Enc.	5/6	? / 2^{66}	Cube-like	[Li+17]
		6/6	$2^{44.8}/2^{128}$	Cond. cube	[CHK22]
		6/6	$\leq \mathbf{2^{40}}$	**Cond. cube**	**Section 3.4**

† stands for "Weak-key subspace", Cond. for conditional.

Table 3.2: Summary of key-recoveries and state-recoveries against Ascon, based on higher-order differentials.

The designers of Ascon also observe that, because public variables (either the nonce in a misuse-free scenario, or the plaintext in a nonce-misuse scenario) are input *on the same row*, they cannot be multiplied together during the first round as Sboxes are applied column-wise. This implies that higher-order differential distinguishers with $2^{r-1}+1$ time and data complexities exist over r rounds, while the algebraic degree is rather bounded by 2^r. In the same spirit, a monomial of degree 2^r with 2^{r-1} public variables $x_0, \ldots, x_{2^{r-1}-1}$ depends on at most 2^{r-1} private variables $k_0, \ldots, k_{2^{r-1}-1}$. In the case of round-reduced Ascon, with *well-chosen fixed public variables* $x_0, \ldots, x_{2^{r-1}-1}$, the set of possible private variables is known and is of size 2^{r-1}. This implies that the coefficient $a_u(k)$ of x^u in any coordinate depends only on a fraction of the key bits. This leads to the recovery of the whole key in a divide-and-conquer manner for 5- or 6-round initializations presented in [Dob+15]. The online phase costs 2^{r-1} in data to recover the value of the derivative. But during the offline phase, this derivative must be computed for each 2^{r-1} possible keys, which leads to a time complexity close to 2^r. In Corollary 3.13, the monomials we consider have a similar property, but depend on 64 private variables. Such an offline phase is therefore too costly.

3.3.2 Conditional cube attacks

Li, Dong & Wang [LDW17] continued analyzing the resistance of Ascon against cube-like attacks by adapting and generalizing the conditional cube attacks against Keccak introduced by Huang, Wang, Xu, Wang & Zhao [Hua+17]. In [LDW17], the authors searched for monomials x^u whose coefficients $(\alpha_{u,i}^{(r)})_{i\in[\![0,63]\!]}$ in all output coordinates after r rounds share a linear divisor L. For initialization reduced to 5 and 6 rounds, they exhibit such monomials *without determining the entire expression of the coefficients*, by analyzing the diffusion of variables throughout the rounds. If such a common divisor L exists, an adversary is able to occasionally deduce its value from the value of the coefficients $(\alpha_{u,i}^{(r)})_{i\in[\![0,63]\!]}$: indeed if there exists i such that $\alpha_{u,i}^{(r)} \neq 0$, it necessarily means that $L \neq 0$.

This attack is extended to 7 rounds (out of 12) in the same paper. In that case, the key space is partitioned into affine spaces $A_{k_0,\dots,k_{s-1}}$ depending on the 0/1 value of some key bits $k_0, \dots, k_{s-1}$. For each space $A_{k_0,\dots,k_{s-1}}$, a family of linear polynomials $L_0, \dots, L_{\ell-1}$ is built. Then, for each $(\varepsilon_0, \dots, \varepsilon_{\ell-1}) \in \mathbb{F}_2^\ell$ a monomial x^u is chosen so that each coefficient $\alpha_{u,i}^{(7)}$ has a linear divisor among $L_0+\varepsilon_0, \dots, L_{\ell-1}+\varepsilon_{\ell-1}$. If the vector $(\alpha_{u,i}^{(7)})_{i\in[\![0,63]\!]}$ is the zero vector, it is assumed[2] that the key lies in $A_{k_0,\dots,k_{s-1}}$ and that the additional conditions $L_i + \varepsilon_i = 0$ for any i are satisfied. Otherwise, if none of the 2^ℓ vectors is the zero vector, it is *guaranteed* that the key does not belong to $A_{k_0,\dots,k_{s-1}}$. This attack however exceeds the 2^{64} data limitation.

Later, Rohit, Hu, Sarkar & Sun *et al.* [Roh+21] presented the first 7-round misuse-free key-recovery cube-attack on Ascon, which does not exceed this limitation. This attack is based on similar observations as the ones made in [Dob+15] which enables the authors to find initial scenarios where coefficients of monomials of degree 64 after 7 rounds only depend on 64 variables. They then precisely analyze the cost of computing and storing all value vectors $(\alpha_{u,i}^{(7)}(k))_{i\in[\![0,63]\!]}$ for all 64-bit partial keys. By *assuming* that the functions $k \mapsto \alpha_{u,i}^{(7)}(k)$ are *all balanced*, this enables them to recover in average a single candidate for the partial key. The other 64-bit partial key is recovered by exhaustive search. They also refine the degree analysis in some initialization scenario to derive new upper-bounds which ultimately lead to cheaper higher-order differential distinguishers.

Rohit & Sarkar [RS21] also tackle weak-key scenarios. By fixing some conditions on the key, the previous bounds on the degree can be lowered. This naturally gives cheaper distinguishers, which however only work if the key is known to satisfy the previous conditions. Stated otherwise, this distinguishes a key satisfying those conditions from a key that does not satisfy them. By considering all these *conditional* distinguishers, the authors therefore obtain a weak-key recovery, by successively trying all distinguishers for the different key conditions.

Very recently, Hu [Hu24] adapted the 7-round distinguisher from [Roh+21],

[2] This is only assumed because the fact that the key lies in $A_{k_0,\dots,k_{s-1}}$ and that any $L_i + \varepsilon_i = 0$ is only a sufficient condition for the vector to be zero.

which is based on the absence of terms of degree 60 in public variables, into a conditional cube attack against 7-round Ascon. This is done by slightly relaxing the initial scenario so that degree-60 terms can *a priori* appear. Then, key conditions are established to avoid this appearance. Testing this cube enables them to verify whether or not the actual key satisfies these conditions.

Remark 3.9. We would like to emphasize the difference between cube and conditional cube methods. In Table 3.2, it is striking that their complexities greatly differ. Indeed, a conditional cube attack can be seen as an alternative to the costly offline phase of a "standard" cube attack. In a standard cube attack, an adversary first has to compute offline the table of values of the coefficients that are later targeted during the online phase. The computation of this table can be long and is proportional to the memory needed to store the table. However it only has to be done once. Afterwards, the online-time cost (which is proportional to the data complexity) is low. A conditional cube attack offers another trade-off: avoiding the precomputation is possible at the cost of higher data and online-time complexities, and an increased cryptanalysis effort. ▷

3.3.3 Probabilistic higher-order differential studies

Most of the probabilistic higher-order differential studies (against Ascon) fall under the differential case, that is, with respect to derivatives of order 1. Last year however, Hu, Peyrin, Tan & Yap [Hu+23] improved the best distinguishers on 4-, 5- and 6-round initializations, with or without conditions on the key, by looking at probabilistic behaviors of derivatives of order strictly larger than 1. They also consider derivatives along spaces unaligned with the canonical basis. Key-recovery attacks based on these distinguishers are also described in the paper.

All of the aforementioned works study the nonce-respecting scenarios and thus focus on the initialization. In Section 3.4, we take another approach by looking at nonce-misuse attacks. This point of view is motivated by the fact that implementation errors will eventually happen and sometimes with high risk. We indeed show that, if a nonce is reused many times, confidentiality is compromised.

3.4 Practical cube attack against nonce-misused Ascon

This section describes the attack, mounted together with Anne Canteaut & Léo Perrin, that is published at published in the IACR Transactions on Symmetric Cryptology, 2022(4) [BCP22].

This attack breaks confidentiality of the messages encrypted using Ascon-128 and Ascon-80pq. It takes place in a particular scenario and thus does not contradict the security claims of the designers. We start by clarifying this scenario.

3.4.1 Nonce-misuse setting and attack model

Nonce-misuse scenario. The nonce-misuse scenario assumes, *contrary to the recommendations of the designers*, that a key/nonce pair is reused several times to encrypt plaintexts. As shown in Figure C.1, the initialization takes as input the key, the nonce and the fixed IV to produce the state Σ_{AD}. In the nonce-misuse scenario, this state Σ_{AD} is therefore fixed once and for all. But if no associated data is proccessed, then the state after initialization Σ_{AD} is equal to the state just before encryption Σ_{E}, as depicted in Figure 3.5, and this latter is, again, fixed.

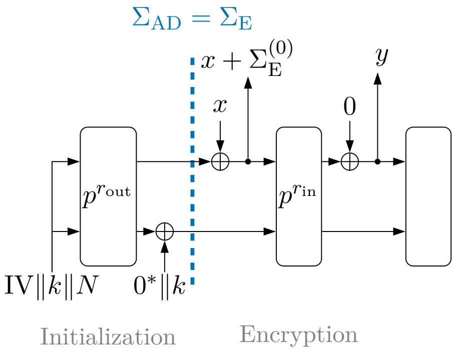


Figure 3.5: Nonce-misuse attack model.

In other words, in the nonce-misuse scenario, with no associated data, the encryption of plaintext blocks only depends on the fixed value Σ_{E}. This is therefore in conflict with the genuine use case in which this value Σ_{E} changes each time a different nonce is used. Such a situation can however occur in cases where Ascon is not properly implemented, or if the attacker gains physical access to the cipher.

Note that the same observation can be made if the *same* associated data is processed before each encryption: in that case Σ_{E}, which depends on key, nonce, and associated data is once again fixed. To simplify, we keep considering the situation depicted in Figure 3.5.

State-recovery attack. In such a situation, the recovery of the full state Σ_{E} is sufficient to break confidentiality of all the messages that have been encrypted with the same key, nonce (and associated data). Indeed, because the permutation p is publicly known, any intermediate state between Σ_{AD} and Σ_{F} (the internal state before the finalization phase) can be computed from the knowledge of Σ_{E} (or any other full state). Any plaintext block x can therefore be recovered by computing $x = y + \Sigma^{(0)}$ where y is the corresponding ciphertext block and $\Sigma^{(0)}$ the first row of the corresponding intermediate state. This is the reason why the recovery of the full state Σ_{E} is our target.

To do so, we consider an adversary in a chosen-plaintext scenario. The attacker can therefore ask for the encryption of any plaintext and observe the associated ciphertext.

By querying the encryption of a 64-bit zero block, the adversary immediately recovers the value of the first row of the state before encryption, $\Sigma_{\mathrm{E}}^{(0)}$, which corresponds to the outer state. The real goal is therefore to recover the four rows of the inner state that are denoted, from top to bottom, by a, b, c and d:

$$a := \Sigma_{\mathrm{E}}^{(1)}, \quad b := \Sigma_{\mathrm{E}}^{(2)}, \quad c := \Sigma_{\mathrm{E}}^{(3)}, \quad d := \Sigma_{\mathrm{E}}^{(4)}.$$

The attack setting. To do so, the following attack only relies on the encryption of messages which are 2-block long, as shown in Figure 3.5. The first block, that is denoted by $x = (x_0, \dots, x_{63}) \in \mathbb{F}_2^{64}$ can take different values. The second one is always the 64-bit zero block.

This way, the corresponding ciphertext is also 2-block long. The first block is of no use, as it corresponds to the sum $x + \Sigma_{\mathrm{E}}^{(0)}$ which only gives information about the already-known outer state $\Sigma_{\mathrm{E}}^{(0)}$. The second one however, that is denoted by y, can be expressed polynomially as a function with 256 private variables, 64 public variables and 64 coordinates. This function is nothing else than p^6 where the output is restricted to the 64 first coordinates, and where the input variables of the first row are public, and the other ones are private. Indeed, if we denote by $p^6_{[\![0,63]\!]} : \mathbb{F}_2^{320} \to \mathbb{F}_2^{64}$ the function defined by:

$$p^6_{[\![0,63]\!]} : X \mapsto \left(p_0^6(X), \dots, p_{63}^6(X)\right),$$

then $p^6_{[\![0,63]\!]}(x, a, b, c, d) = y$. Section 3.4.2 is dedicated to the study of the algebraic properties of this function, that will be needed to mount the state-recovery attack.

Comparison with [CHK22]. It is worth noting that our attack recovers the whole state Σ_{E} as soon as enough plaintexts of the previous form are encrypted from the same internal state Σ_{E}, *i.e.*, from the same triple of key, nonce and associated data. This differs from the attack scenario in the work by Chang, Hong & Kang [CHK22] which recovers a part of the state Σ_{E} only if it satisfies a few conditions. It follows that the first step of their attack needs to be repeated for 32 triples of key, nonce and associated data in average, until the corresponding state can be recovered.

Comparison with [VV18]. There exists a generic nonce-misuse attack against the Sponge Duplex construction that is presented by Vaudenay & Vizár [VV18]. This attack works as follows. First, by asking for the encryption of the 64-bit zero block, the first outer state is recovered and therefore the first plaintext block $P^{(0)}$ corresponding to any ciphertext obtained with the same key, nonce and IV. Then by asking for the encryption of the 2-block message $(P^{(0)}, 0)$, the second ciphertext block exactly corresponds to the second value of the outer state, and the second plaintext block can therefore be recovered. The attacker continues asking the decryption of messages $(P^{(0)}, \dots, P^{(\ell-1)}, 0)$ until the full plaintext has been recovered.

This attack is cheap as a single adapted query is necessary for each block that needs to be decrypted. It has then a lower query complexity than our attack for messages whose length does not exceed 2^{40} blocks. However, our attack and the generic attack differ both in their settings and their intention: contrary to [VV18], the attack that is presented in Section 3.4 relies on the actual permutation of Ascon that is used in the mode. It therefore gives new insights about its security. Moreover, once the state is recovered with our attack, the confidentiality of any message encrypted with the same key/nonce/associated data triplet is compromised. In particular, it can be applied to previous or future encryption with the same triple without any other interaction with the cipher.

Toward key-recovery? Finally, it is tempting to mount an actual key-recovery from the state recovery that is described in the next section. This does not seem as simple as one could think. Contrary to the genuine MonkeyDuplex [Ber+12b] or SpongeWrap [Ber+12a] constructions, in the AEAD mode of Ascon, the permutation used after initialization and before finalization are keyed permutations. They are obtained using feed-forward key additions as depicted in Figure C.1. This in particular means that a state-recovery does not directly lead to a key-recovery (by going through the initialization backward) or a tag forgery (by going through the finalization). However in the case of Ascon-80pq *only*, as pointed out by Chang, Hong & Kang [CHK22], a state-recovery as described in Section 3.4 can lead to a key-recovery of the 160-bit key in less than 2^{160} operations. Nevertheless, this requires a nonce-misuse setting, and more operations than the 2^{128} claimed by the designers for the nonce-respecting setting.

Now that both the objective and the scenario in which the attack takes place are clarified, we start describing its actual setup by making observations on the algebraic properties of the function $\mathcal{E}$.

3.4.2 Algebraic properties of p^6 in a nonce-misuse setting

As required by the nonce-misuse scenario, we study in this section the properties of p^6 used with public $x_i, i \in [\![0, 63]\!]$ and private variables $a_i, b_i, c_i, d_i, i \in [\![0, 63]\!]$. As explained in Section 3.1.2.b, in order to mount a cube attack, it is necessary to find some monomial x^u, $u \in \mathbb{F}_2^{64}$ in the ANF of the output bits of p^6 such that the family $(\alpha_{u,i})_{i \in [\![0,63]\!]}$ of coefficients (viewed as polynomials $\alpha_{u,i} \in \mathbb{F}_2[a, b, c, d]$) is simple enough to recover partial information about a, b, c, d.

To do so, we use the same heuristic as the one presented in the mentioned section: a coefficient $\alpha_{u,i}$ in variables a, b, c, d has a bigger chance to be simple if its corresponding monomial x^u is of high degree in variables $(x_i)_{i \in [\![0,63]\!]}$. We therefore study the terms $\alpha_{u,i} x^u$ that appear in the successive ANF of p^r for $r \in [\![0, 6]\!]$ where x^u is of high degree. More precisely, for any $r \in [\![0, 6]\!]$ let us express the ANF of p^r as:

$$\forall i \in [\![0, 319]\!], \quad (p^r)_i \colon (x, a, b, c, d) \mapsto \sum_{u \in \mathbb{F}_2^n} \alpha_{u,i}^{(r)} x^u,$$

where $\alpha_{u,i}^{(r)}$ is a polynomial of $\mathbb{F}_2[a, b, c, d]$ for any u, i, r.

A *highest-degree term* refers to $\alpha_{u,i}^{(r)} x^u$ where $\alpha_{u,i}^{(r)}$ is a non-zero polynomial and x^u is of highest possible degree among all terms that appear in p^r. Such $\alpha_{u,i}^{(r)}$ is called a *highest-degree coefficient.*

Remark 3.10. We do not take into account the degree in the variables a_i, b_i, c_i, d_i for now. Thus, *if not stated otherwise*, by degree, we mean algebraic degree in the variables $(x_i)_{i \in [\![0,63]\!]}$, and by constant coefficients we mean $\alpha_{0,i}^{(r)} \in \mathbb{F}_2[a, b, c, d]$. ▷

We start by presenting the following simple yet very important observation about the *highest-degree* terms in the successive ANFs of p^r.

Proposition 3.11 (Highest-degree terms in the ANF of p^r). *Let $r \in [\![1, 6]\!]$. Then:*

- *(Bound on the degree)* $\forall\ i \in [\![0, 319]\!], \forall\ u \in \mathbb{F}_2^{64}, \alpha_{u,i}^{(r)} \neq 0 \implies \deg_a(x^u) \leq 2^{r-1}$.
- *(Tight bound)* $\exists\ i \in [\![0, 319]\!], \exists\ u \in \mathbb{F}_2^{64}, \alpha_{u,i}^{(r)} \neq 0$ *and* $\deg_a(x^u) = 2^{r-1}$.
- *(Trails of highest-degree terms) Let $r \geq 2$. Let $i \in [\![0, 319]\!], u \in \mathbb{F}_2^{64}$ such that $\alpha_{u,i}^{(r)} \neq 0$, and $\deg_a(x^u) = 2^{r-1}$. Then $\alpha_{u,i}^{(r)}$ can be expressed as a sum $\alpha_{u,i}^{(r)} = \sum \alpha_{v,j}^{(r-1)} \alpha_{w,\ell}^{(r-1)}$ over some v and w with $\deg_a(x^v) = \deg_a(x^w) = 2^{r-2}$.*

Proof. At the input of the first Sbox layer $p_S^{(1)}$, each public variable x_i only appears in the i-th coordinate of the first row of the state. In particular, public variables cannot be multiplied together during the first round, as the Sbox is the only non-linear operation and it is applied column-wise. So after one round, the highest-degree in the ANF of p^1 is still $1 = 2^{1-1}$. Afterwards, because the Sbox is quadratic, the degree in the variables $(x_i)_{i \in [\![0,63]\!]}$ can at most double. This proves the announced bound.

Regarding the existence of some terms $\alpha_{u,i}^{(r)} x^u$ with $\alpha_{u,i}^{(r)} \neq 0$ and $\deg_a(x^u) = 2^{r-1}$, such terms will be exhibited in Section 3.4.3 for $r = 6$ so the bound is tight for 6 rounds. But, the existence of a term of degree 32 at round 6 implies the existence of terms of degree 2^{r-1} at each round $r \in [\![1, 6]\!]$, otherwise the degree should grow more than twice during one round, which is excluded by the quadraticity of the round function.

Finally, the last statement only expresses the fact that monomials of highest degree x^u can only be obtained as products of monomials of highest degree one round before. Indeed, degree 2^{r-1} can be reached by multiplying monomials of degree at most 2^{r-2}. □

Remark 3.12. The previous result ensures the existence of a non-zero *polynomial* $\alpha_{u,i}^{(r)}$. However the value of this polynomial, can, and does change in practice with the value of (a, b, c, d), and therefore depends on the key and the nonce. We do not ensure here the existence of a coefficient that is constant independently of the key/nonce pair. This would help to distinguish the function, but probably not to recover an equation in private variables.

Furthermore, according to our observations, it seems very likely that an even stronger second statement holds. Indeed, the bound seems to be tight for every coordinate $i \in [\![0, 319]\!]$. Furthermore it also seems that for every value $(a, b, c, d) \in \mathbb{F}_2^{256}$, and for every coordinate i, there exists a highest-degree coefficient $\alpha_{u,i}^{(r)}$ such that $\alpha_{u,i}^{(r)}(a, b, c, d) = 1$. We do not need either of these supposed properties in the following. ▷

From Proposition 3.11, we immediately deduce the following corollary.

Corollary 3.13 (Recursive formula for highest-degree coefficients)**.** *Let $r \in [\![1, 6]\!]$. Let u be such that $\deg_a(x^u) = 2^{r-1}$. Let $i \in [\![0, 319]\!]$ such that $\alpha_{u,i}^{(r)} \neq 0$. Then $\alpha_{u,i}^{(r)}$ can be expressed as a sum of products, where each product has the following form:*

$$\prod_{j \in \mathrm{Supp}(u)} L_j, \quad \textit{where } L_j \in \{a_j + 1,\ 1,\ c_j + d_j + 1,\ a_j\}\,.$$

Proof. By inductively using the third statement of Proposition 3.11, it is clear that any highest-degree coefficient at round r can be expressed as a sum of products of 2^{r-1} highest-degree coefficients which appear in the ANF of p^1, that is, coefficients of monomials of degree $2^{1-1} = 1$ in p^1. The ANF of $p_S^{(1)}$ is given in Table 3.3. Let $j \in [\![0, 63]\!]$. The monomial x_j is present only in the j-th column, with four possible coefficients depending on the row: $a_j + 1, 1, c_j + d_j + 1$ and a_j. This remains true for the ANF of $p_S^{(1)} \circ p_C^{(1)}$ as the composition of the input with $p_C^{(1)}$ only amounts to change some variables b_j by $b_j + 1$, which does not impact the previous coefficients. After $p_S^{(1)} \circ p_C^{(1)}$, there are then at most one monomial x^j on each row. The linear layer, that is applied row-wise (see Section 3.2.3), can therefore only *copy* this single monomial x_j in other coordinates, but not combine linearly two such monomials. Therefore, a non-zero coefficient of x_j in the ANF of $p^1 = p_L^{(1)} \circ p_S^{(1)} \circ p_C^{(1)}$ can only be $a_j + 1, 1, c_1 + d_j + 1$ or a_j. □

Initial state	$S(x_j, a_j, b_j, c_j, d_j)$		
x_j	$(a_j + 1)x_j$	$+$	$a_jb_j + a_jd_j + a_j + b_j + c_j$
a_j	x_j	$+$	$a_jb_j + a_jc_j + b_jc_j + a_j + b_j + c_j + d_j$
b_j	0	$+$	$c_jd_j + a_j + b_j + d_j + 1$
c_j	$(c_j + d_j + 1)x_j$	$+$	$a_j + b_j + c_j + d_j$
d_j	a_jx_j	$+$	$a_jd_j + a_j + c_j + d_j$

Table 3.3: ANF of Column j after initialization and after the first Sbox layer $p_S^{(1)}$.

In the following, for any $j \in [\![0, 63]\!]$, we denote by e_j the linear combination $c_j + d_j + 1 =: e_j$. With this notation, any highest-degree coefficient $\alpha_{u,i}^{(r)}$ for some u, i, r only depends on a_j and e_j with $j \in \mathrm{Supp}(u)$. In particular, any cube attack targeting highest-degree coefficients can at best lead to the recovery of a and $e := c + d$. In order to mount a full recovery, we thus need to focus on other coefficients. In our case, we consider *sub-leading terms*, that is terms $\alpha_{u,i}^{(r)}x^u$ for which $\alpha_{u,i}^{(r)} \neq 0$ and $\deg_{\mathrm{a}}(x^u) = 2^{r-1} - 1$. We refer to the corresponding coefficient $\alpha_{u,i}^{(r)}$ and monomial x^u as *sub-leading coefficient* and *sub-leading monomial.* The same kind of observations as the ones made in Proposition 3.11 and Corollary 3.13 can be done for sub-leading terms.

Proposition 3.14 (Sub-leading terms in the ANF of p^r)**.** *Let* $r \in [\![2,6]\!]$*. Let* $i \in [\![0,319]\!]$, $u \in \mathbb{F}_2^{64}$ *such that* $\alpha_{u,i}^{(r)} \neq 0$ *and* $\deg_a(x^u) = 2^{r-1} - 1$*. Then* $\alpha_{u,i}^{(r)}$ *can be expressed as a sum* $\alpha_{u,i}^{(r)} = \sum \alpha_{v,j}^{(r-1)} \alpha_{w,\ell}^{(r-1)}$ *over some* v *and* w *corresponding to monomials which satisfy one of the two following conditions:*

- *either* $\deg_a(x^v) = 2^{r-2}$ *and* $\deg_a(x^w) = 2^{r-2} - 1$ *(or the other way around);*
- *or,* $\deg_a(x^v) = \deg_a(x^w) = 2^{r-2}$ *and* $\gcd(x^v, x^w) = x_j$ *for some* $j \in [\![0,63]\!]$*.*

Proof. A term of degree $\alpha_{u,i}^{(r)}x^u$ of degree $\deg_{\mathrm{a}}(x^u) = 2^{r-1} - 1$ in the ANF of r rounds is necessarily obtained by a *product* of terms of the ANF of $r-1$ rounds. Indeed, the degree of the ANF of $r-1$ rounds is upper-bounded by 2^{r-2}. Let $v, w \in \mathbb{F}_2^{64}$ be such that $\mathrm{wt}(v), \mathrm{wt}(w) \leq 2^{r-2}$ and $\mathrm{wt}(v+w) = 2^{r-1} - 1$, it must hold that:

$$\mathrm{wt}(v) + \mathrm{wt}(w) - |\mathrm{Supp}(v) \cap \mathrm{Supp}(w)| = \mathrm{wt}(v+w) = 2^{r-1} - 1.$$

If $\mathrm{wt}(v) < 2^{r-2} - 1$, then

$$\begin{aligned}\mathrm{wt}(v) + \mathrm{wt}(w) - |\mathrm{Supp}(v) \cap \mathrm{Supp}(w)| &< 2^{r-2} - 1 + 2^{r-2} - |\mathrm{Supp}(v) \cap \mathrm{Supp}(w)| \\ &\leq 2^{r-1} - 1.\end{aligned}$$

So it must hold that $\mathrm{wt}(v), \mathrm{wt}(w) \geq 2^{r-2} - 1$. If $\mathrm{wt}(v) = \mathrm{wt}(w) = 2^{r-2} - 1$, then:

$$\begin{aligned}\mathrm{wt}(v) + \mathrm{wt}(w) - |\mathrm{Supp}(v) \cap \mathrm{Supp}(w)| &= 2^{r-1} - 2 - |\mathrm{Supp}(v) \cap \mathrm{Supp}(w)| \\ &< 2^{r-1} - 1.\end{aligned}$$

So the only remaining possibilities are $\mathrm{wt}(v) = 2^{r-2}$ and $\mathrm{wt}(w) = 2^{r-2} - 1$ (which implies $|\mathrm{Supp}(v) \cap \mathrm{Supp}(w)| = 0$), the symmetric possibility and $\mathrm{wt}(v) = 2^{r-2}$ and $\mathrm{wt}(w) = 2^{r-2}$, which implies $|\mathrm{Supp}(v) \cap \mathrm{Supp}(w)| = 1$. □

Corollary 3.15 (Recursive formula for sub-leading coefficients)**.** *Let* $r \in [\![2,6]\!]$*. Let* u *be such that* $\deg_a(x^u) = 2^{r-1} - 1$*. Let* $i \in [\![0,319]\!]$ *be such that* $\alpha_{u,i}^{(r)} \neq 0$*. Then* $\alpha_{u,i}^{(r)}$ *can be expressed as a sum of products of* 2^{r-1}*coefficients of terms of the ANF of* $p_C^{(2)} \circ p$*.*

Let $\prod_{(v,j) \in Z} \alpha_{v,j}$ *be one of these products. Then it satisfies one of the following conditions:*

- $\forall (v,j) \in Z$, $\deg_a(x^v) = 1$*. Furthermore there exist* $(v,j), (w,k) \in Z$*, such that* $v = w$ *and* $j \neq k$ *(i.e. only coefficients of monomials of degree 1, with two (possibly different) coefficients of the same monomial).*

- *or,* $\exists!\ (v,j) \in Z$*,* $v = 0$ *and* $\forall (w,k) \in Z \setminus \{(v,j)\}\ \deg_a(x^v) = 1$ *(i.e.* $2^{r-1} - 1$ *coefficients of monomials of degree 1, and one constant coefficient).*

Proof. This is a direct consequence of Proposition 3.14 and the proof is similar to the one of Corollary 3.13. The only difference is that the ANF of $p_C^{(2)} \circ p^{(1)}$ is considered rather than the one of $p^{(1)}$. The reason is that sub-leading coefficients after one round correspond to the constant coefficients $\alpha_{0,i}^{(1)}$ with $i \in [\![0, 319]\!]$. Some of these coefficients are flipped through $p_C^{(2)}$. □

In the following, we only focus on highest-degree and sub-leading terms. Regarding highest-degree terms, they benefit from the inherent symmetries due to the structure of the permutation of Ascon. In the following, we denote by $\lll$ the cyclic shift on 64-bit words. In other words, for any $u \in \mathbb{F}_2^{64}$ and $\ell \in [\![0, 63]\!]$, we denote by $u \lll \ell \in \mathbb{F}_2^{64}$ the element defined by the following relation:

$$\forall\ i \in [\![0, 63]\!]\,, \quad i \in \mathrm{Supp}(u \lll \ell) \iff (i - \ell \bmod 64) \in \mathrm{Supp}(u).$$

Proposition 3.16 (Rotation invariance for highest-degree terms)**.** *Let* $r \in [\![1, 6]\!]$*. Let* $j \in [\![0, 4]\!]$*, let* $u \in \mathbb{F}_2^{64}$ *such that* $\deg_a(x^u) = 2^{r-1}$*. Let* $\alpha_{u,0+64j}^{(r)}$ *be decomposed as:*

$$\alpha_{u,0+64j}^{(r)} = \sum_{v_a, v_b, v_c, v_d \in \mathbb{F}_2^{64}} \beta_{a,b,c,d}\ a^{v_a} b^{v_b} c^{v_c} d^{v_d},$$

with $\beta_{a,b,c,d} \in \mathbb{F}_2$*. Then for any* $i \in [\![0, 63]\!]$*, we have:*

$$\alpha_{u \lll \ell, i+64j}^{(r)} = \sum_{v_a, v_b, v_c, v_d \in \mathbb{F}_2^{64}} \beta_{a,b,c,d}\ a^{v_a \lll \ell} b^{v_b \lll \ell} c^{v_c \lll \ell} d^{v_d \lll \ell}.$$

Proof. The only layers that break the symmetries in the ANF of Ascon are the constant additions $p_C^{(r)}$, $r \in [\![1, 6]\!]$. But constant additions only modify the polynomial expression of the constant coefficients. Therefore the only constant addition that can influence the expressions of highest-degree terms is $p_C^{(0)}$. But it is already noticed in the proof of Corollary 3.13 that it is not the case. □

Proposition 3.16 then enables us to only study highest-degree terms in a single column, as any property can be adapted to any other column. This is not the case for sub-leading terms which are influenced by the first two constant additions. The first one, $p_C^{(0)}$, can completely be ignored as it only flips the value of a few unknown bits b_i into $b_i + 1$. Recovering the value of ones or the others is equivalent, up to a flip at the end of the state-recovery. The second one, $p_C^{(1)}$, flips the value of a few constant terms $\alpha_{0,i}^{(1)}$. The equations recovered in Section 3.4.4.c may be based on them but, as we will see, these flips will have no influence on the hardness of solving such equations.

3.4.3 Two specific families of highest-degree terms

For now, we focus on exhibiting conditional cubes based on highest-degree terms. To do so, we first fix as *primary variable* x_0 so that all described monomials x^u now always contain x_0. Because of Proposition 3.16, every statement is easily generalized to any other choice of primary variable by a mere shift of all indices.

Linear divisors after 2 rounds. Let $i \in [\![0, 319]\!]$, $\ell \in [\![1, 63]\!]$. Let consider $x_0 x_\ell$, that we denote by x^u (with $u = \xi^{(0)} + \xi^{(\ell)}$), and its coefficients $\alpha_{u,i}^{(2)} \neq 0$ in the ANF of $p_i^{(2)}$. By Corollary 3.13, $\alpha_{u,i}^{(2)}$ can be decomposed as:

$$\alpha_{u,i}^{(2)} = \sum_{(j,k) \in Z} \alpha_{\xi^{(0)},j}^{(1)} \alpha_{\xi^{(\ell)},k}^{(1)},$$

over a set of pairs Z. For any $(j, k) \in Z$, let us decompose j as $j = j_c + 64 \cdot j_r$ with $j_c \in [\![0, 63]\!]$ its row index and $j_c \in [\![0, 4]\!]$ its column index. As shown by Table 3.3, the coefficients of x_0 after one round are only row-dependent and are linear for rows $0, 3, 4$. Therefore, if for all $(j, k) \in Z$, j_r is *constant and equal to* $0, 3$ or 4, then $\alpha_{u,i}^{(2)}$ has a linear divisor which belongs to $\{a_0 + 1, e_0, a_0\}$.

This phenomenon does happen in the ANF of $p^{(2)}$. More importantly, for some ℓ, this happens for *each* non-zero $\alpha_{u,i}^{(2)}$ and the linear divisor is sometimes *independent* of i. In that case, the family $(\alpha_{u,i}^{(2)})_{i \in [\![0,319]\!]}$ has a common linear divisor.

We therefore introduce the subsets $Z_{a_0+1}, Z_{a_0}, Z_{e_0}, Z_0, Z'$ that partition $[\![1, 63]\!]$ and that are defined as follows. Let $\ell \in [\![1, 63]\!]$, let $u = \xi^{(0)} + \xi^{(\ell)}$ so that $x^u = x_0 x_\ell$. Then:

- $\ell \in Z_{a_0+1}$ if $a_0 + 1$ is a linear common divisor of the coefficients $(\alpha_{u,i}^{(2)})_{i \in [\![0,319]\!]}$.
- $\ell \in Z_{a_0}$ if a_0 is a linear common divisor of the coefficients $(\alpha_{u,i}^{(2)})_{i \in [\![0,319]\!]}$.
- $\ell \in Z_{e_0}$ if e_0 is a linear common divisor of the coefficients $(\alpha_{u,i}^{(2)})_{i \in [\![0,319]\!]}$.
- $\ell \in Z_0$ if $\forall\, i \in [\![0, 319]\!]$, $\alpha_{u,i}^{(2)} = 0$.
- $\ell \in Z'$ otherwise.

This partition classifies the monomials $x_0 x_\ell$ in the ANF of $p^{(2)}$. Because its ANF is still sparse, the contents of the sets $Z_{a_0+1}, Z_{a_0}, Z_{e_0}, Z_0, Z'$ can be clearly identified and are detailed in Table 3.4.

Set	Cardinality
$Z_{a_0+1} = \{9, 12, 18, 19, 21, 28\}$	6
$Z_{a_0} = \{7, 24, 41, 43, 52\}$	5
$Z_{e_0} = \{17, 35, 40, 46, 55\}$	5
$Z_0 = \left\{ \begin{array}{l} 1, 4, 5, 6, 8, 14, 15, 16, 26, 27, 30, 34, \\ 37, 38, 48, 49, 50, 56, 58, 59, 60, 63 \end{array} \right\}$	22
$Z' = \left\{ \begin{array}{l} 2, 3, 10, 11, 13, 20, 22, 23, 25, 29, 31, 32, 33, \\ 36, 39, 42, 44, 45, 47, 51, 53, 54, 57, 61, 62 \end{array} \right\}$	25

Table 3.4: The sets $Z_{a_0+1}, Z_{a_0}, Z_{e_0}, Z_0, Z'$.

Two classes of highest-degree terms. From there, we exhibit two classes of highest-degree monomials after 6 rounds whose coefficients have a very particular structure. In order to build monomials of degree 32, 31 variables have to be chosen, as we already (arbitrarily) selected x_0.

Proposition 3.17 (Class $\mathcal{C}(a_0 + 1, e_0)$)**.** *Let $Z \subset Z_{a_0+1} \cup Z_{e_0} \cup Z_0$ be such that $|Z| = 31$. Let $u \in \mathbb{F}_2^{64}$ be such that* $\mathrm{Supp}(u) := \{0\} \cup Z$. *Let $i \in [\![0, 319]\!]$. Then, $\alpha_{u,i}^{(6)}$ can be decomposed as:*

$$\alpha_{u,i}^{(6)} = (a_0 \oplus 1)\gamma_{u,i,a} + e_0\gamma_{u,i,e},$$

for some $\gamma_{u,i,a}, \gamma_{u,i,e} \in \mathbb{F}_2[a, b, c, d]$.

Proof. According to Corollary 3.13, $\alpha_{u,i}^{(6)}$ can be seen as a sum of products $\prod_{(v,j)\in W} \alpha_{v,j}^{(2)}$ where $\forall(v, j) \in W$, $\mathrm{wt}(v) = 2$. For each product, there exists a single (v, j) such that $x^v = x_0x_\ell$ for some $\ell \in Z$. But according to the definitions of Z, all coefficients $(\alpha_{v,k}^{(2)})_{k\in[\![0,319]\!]}$ of $x^v = x_0x_\ell$ are either 0, or divisible by $a_0 + 1$ or e_0. The result follows immediately. □

In the same way, we prove the following proposition for another class of coefficients with a slightly different decomposition.

Proposition 3.18 (Class $\mathcal{C}(a_0, e_0)$)**.** *Let $Z \subset Z_{a_0} \cup Z_{e_0} \cup Z_0$ such that $|Z| = 31$. Let $u \in \mathbb{F}_2^{64}$ such that* $\mathrm{Supp}(u) := \{0\} \cup Z$. *Let $i \in [\![0, 319]\!]$. Then, $\alpha_{u,i}^{(6)}$ can be decomposed as:*

$$\alpha_{u,i}^{(6)} = a_0\varphi_{u,i,a} + e_0\varphi_{u,i,e},$$

for some $\varphi_{u,i,a}, \varphi_{u,i,e} \in \mathbb{F}_2[a, b, c, d]$.

Definition 3.19 (Classes $\mathcal{C}(a_0 + 1, e_0)$ and $\mathcal{C}(a_0, e_0)$)**.** Let $u \in \mathbb{F}_2^{64}$. The monomial x^u is said to belong to the class $\mathcal{C}(a_0 + 1, e_0)$ if u matches the criteria of Proposition 3.17. Similarly x^u belongs to the class $\mathcal{C}(a_0, e_0)$ if u matches the criteria of Proposition 3.18. ▷

Partial linear divisors. As shown by Propositions 3.17 and 3.18, the coefficients of monomials in $\mathcal{C}(a_0+1, e_0)$ and $\mathcal{C}(a_0, e_0)$ are highly structured. This structure generalizes the idea of coefficients with linear divisors, with instead two partial linear divisors. This generalization is needed as the same reasoning cannot lead to highest-degree coefficients that share a common divisor. Indeed for any $\beta \in \{a_0+1, a_0, e_0\}$, we observe that $|Z_0 \cup Z_\beta| < 31$. Interestingly, linear common divisors exist if the number of rounds is reduced or when looking at the initialization of Ascon. As an example, Li, Dong & Wang [LDW17] mounted conditional cube attacks against 5- and 6-round initializations. Their choice of coefficients that share a linear divisor can be explained by such a reasoning.

Two specific choices of monomials. In the following, we leverage the structure of the coefficients of monomials in classes $\mathcal{C}(a_0+1, e_0)$ and $\mathcal{C}(a_0, e_0)$ to mount our cube attack. To do so, we only need *one specific monomial of each class.* These monomials are chosen by following a simple rationale.

For the monomial $x^v \in \mathcal{C}(a_0+1, e_0)$, we build the set of indices Z by selecting all indices of Z_{e_0}, all indices of Z_0 and the 4 smallest indices of Z_{a_0+1}. Similarly, the monomial $x^w \in \mathcal{C}(a_0, e_0)$ is built by selecting all indices of Z_{e_0}, all indices of Z_0 and the 4 smallest indices of Z_{a_0}. The supports of v, w are detailed in Table 3.5.

	Prim.[†]	Z_{a_0+1}	Z_{a_0}	Z_{e_0}	Z_0	Z'
v	0	9, 12, 18, 19	-	17, 35, 40, 46, 55	1, 4, 5, 6, 8, 14, 15, 16, 26, 27, 30, 34, 37, 38, 48, 49, 50, 56, 58, 59, 60, 63	-
w	0	-	7, 24, 41, 43	17, 35, 40, 46, 55	1, 4, 5, 6, 8, 14, 15, 16, 26, 27, 30, 34, 37, 38, 48, 49, 50, 56, 58, 59, 60, 63	-

Table 3.5: Supports of v and w. [†] Prim. stands for primary variable.

For the inner-state recovery, we also need shifted versions of x^v, x^w, namely $x^{v \lll j}, x^{w \lll j}$ for any $j \in [\![0, 63]\!]$. Each of them belongs to a class $\mathcal{C}(a_j+1, e_j)$ or $\mathcal{C}(a_j, e_j)$ for $j \in [\![0, 63]\!]$, that can be defined as in Definition 3.19. Their coefficients have thus a decomposition similar to the ones described in Propositions 3.17 and 3.18.

3.4.4 Inner-state recovery against the full encryption of Ascon

Everything is now setup to describe the actual recovery of the inner state. This attack considers an *adaptative chosen-plaintext scenario* and is decomposed into three steps, that will be successivelly described in this section.

1. The first step, that will be described in Section 3.4.4.a, recovers all e_j and (in average) half of the bits a_j, from the values $(\alpha^{(6)}_{u,i})_{i \in [\![0,63]\!]}$ where $u = v \lll j$ or $u = w \lll j$ for all $j \in [\![0, 63]\!]$. These bits e_j, a_j are deduced from the general

form of the coefficients exhibited in Propositions 3.17 and 3.18 even if their expressions will not be explicitly computed.

2. The second step (see Section 3.4.4.b) recovers the remaining a_j by a classical cube-attack targeting other highest-degree monomials x^u. Indeed, the bits recovered at Step 1 drastically simplify the polynomial expressions of the considered $(\alpha^{(6)}_{u,i})_{i\in[\![0,63]\!]}$, which can this time be exactly computed.
3. Finally, the third step (see Section 3.4.4.c) recovers most of the bits b_j and c_j by targeting some sub-leading monomials, whose coefficients are sparse polynomials of degree at most 2 in these unknowns. The few remaining bits b_j and c_j are eventually recovered by an exhaustive search.

3.4.4.a First step: recovering most of the bits a_i and e_i

In the first step, we recover the value of e_j for all j and the value of a_j for most of the indices j. To do so, we mount a conditional cube attack using x^v and x^w defined in Table 3.5, and their 63 rotations $x^{v\lll j}, x^{w\lll j}$ with $j \in \{1, \dots, 63\}$.

The case of a_0, e_0. For now, let us focus on $j = 0$, that is, on x^v. For any $i \in [\![0, 319]\!]$, the algebraic expression of $\alpha_{v,i}$ is partially known from Proposition 3.17. However, as shown on Figure 3.5, only $p^{(6)}_{[\![0,63]\!]}(x, a, b, c, d)$ is accessible to an attacker. Therefore, only $(\alpha^{(6)}_{v,i})_{i\in[\![0,63]\!]}$ are considered and the values of these coefficients are recovered using Eq. (3.2), or equivalently using Proposition 2.10. As evaluating the 64 coefficients corresponds to evaluating the derivative of 64 coordinates *along the same direction*, these evaluations are done in parallel for a time and data complexity of 2^{32}, by computing $\varepsilon := \sum_{x\preceq v} p^6_{[\![0,63]\!]}(x, a, b, c, d)$.

The adversary then studies the obtained system of equations:

$$\forall\ i \in [\![0, 63]\!]\,, \quad (a_0 \oplus 1)\gamma_{u,i,a} + e_0\gamma_{u,i,e} = \varepsilon_i.$$

If there exists i such that $\varepsilon_i = 1$, then necessarily $a_0 = 0$ or $e_0 = 1$. Without further knowledge on $\gamma_{v,i,a}, \gamma_{v,i,e}$ for all i, this is *in theory* the only way of recovering information. However, experimental results show that the reciprocal statement is *in practice* satisfied. In other words, each time that $\varepsilon_i = 0$ for all $i \in [\![0, 63]\!]$, then $a_0 = 1$ and $e_0 = 0$ can be assumed. This is highlighted by Figure 3.6. As expected if $a_0 = 1, e_0 = 0$, any observed value for $(\alpha^{(6)}_{v,i})_{i\in[\![0,63]\!]}$ is necessarily[3] 0. But more importantly, in the three remaining cases the coefficients $(\alpha^{(6)}_{v,i})_{i\in[\![0,63]\!]}$ behave randomly: each individual $\alpha^{(6)}_{v,i}$ not only follows a Bernouilli distribution with probability 0.5 (see the bottom part of Figure 3.6), but they also *seem* to behave independently from one another. This is highlighted by the top of Figure 3.6, where the vector $(\alpha^{(6)}_{v,i})_{i\in[\![0,63]\!]}$ seems to follow a binomial distribution with success probability 0.5.

[3] This can be thought as a *conditional distinguisher*: under the assumption, $a_0 = 1, e_0 = 0$, the value of the coefficients is constant.

This indicates that if $(a_0, e_0) \neq (1, 0)$, the vector $(\alpha^{(6)}_{v,i})_{i\in[\![0,63]\!]}$ is the zero vector with an extremely low probability. During our experimentation, that we expect to be representative due to the 4000 trials, this event was never observed. Therefore, when the zero vector is observed, the guess that $a_0 = 1$ and $e_0 = 0$ is (almost) never a wrong guess.

The same observations are made for the case of x^w with Figure 3.7. In that case, $(\alpha^{(6)}_{w,i})_{i\in[\![0,63]\!]}$ is zero for $a_0 = 0, e_0 = 0$ and behaves randomly in the three other cases. In the light of these experimental results, the following two assumptions are made in the remaining of this section.

Assumption 3.20. *Let ε be the vector of values of $(\alpha^{(6)}_{v,i})_{i\in[\![0,63]\!]}$ computed during the cube attack. If ε is the zero vector, then the guess $(a_0 \oplus 1 = 0$ and $e_0 = 0)$ is wrong with a negligible probability.*

Assumption 3.21. *Let ε be the vector of values of $(\alpha^{(6)}_{w,i})_{i\in[\![0,63]\!]}$ computed during the cube attack. If ε is the zero vector, then the guess $(a_0 = 0$ and $e_0 = 0)$ is wrong with a negligible probability.*

The general case. The index of the considered bits a_0, e_0 is the index of the primary variable x_0. But according to Proposition 3.16, everything remains identical if we choose another primary variable. So, Assumptions 3.20 and 3.21 can be adapted to the shifted monomials $x^{v\lll j}, x^{w\lll j}$ for any $j \in [\![1, 63]\!]$. This enables us to recover all the bits e_i and, in average, half of the bits a_i by following Algorithm 1.

Algorithm 1 Step 1: v and w are defined in Table 3.5.

```
Output: $e_j$ for all $j \in \{0,\dots,63\}$ and $a_j$ for some $j \in \{0,\dots,63\}$
  for all $j \in \{0,\dots,63\}$ do
    $a_j \leftarrow -1, e_j \leftarrow -1$                          ▷ Initialize all variables.
  end for

  for all $j \in \{0,\dots,63\}$ do
    $\varepsilon_v \leftarrow$ CubeSumVector($x^{v\lll j}$)
    if $\varepsilon_v = (0,\cdots,0)$ then
      $a_j \leftarrow 1, e_j \leftarrow 0$                           ▷ Assumption 3.20
    else
      $\varepsilon_w \leftarrow$ CubeSumVector($x^{w\lll j}$)
      if $\varepsilon_w = (0,\cdots,0)$ then
        $a_j \leftarrow 0, e_j \leftarrow 0$                         ▷ Assumption 3.21
      else
        $e_j \leftarrow 1$                                           ▷ No assumption
      end if
    end if
  end for
```

The cost of this first step is thus upper-bounded by 64 × 2 = 128 computations of derivatives along spaces of dimension 32. In other words, time and data costs

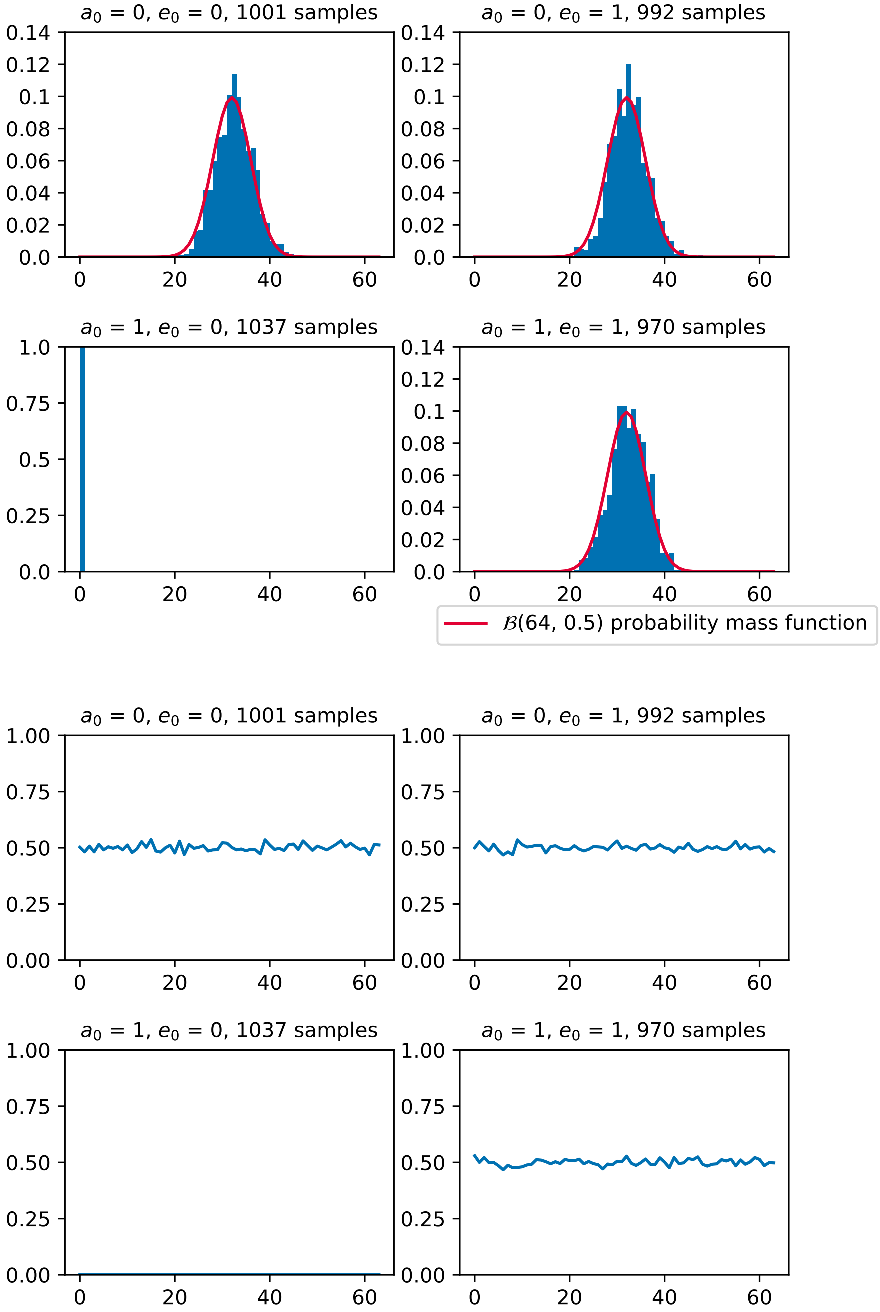


Figure 3.6: Distribution of the Hamming weight of $(\alpha^{(6)}_{v,i})_{i\in[\![0,63]\!]}$ (top) and balancedness of each $\alpha^{(6)}_{v,i}$ for any $i \in [\![0, 63]\!]$ (bottom) for 4000 random inner states sorted according to the value of (a_0, e_0).

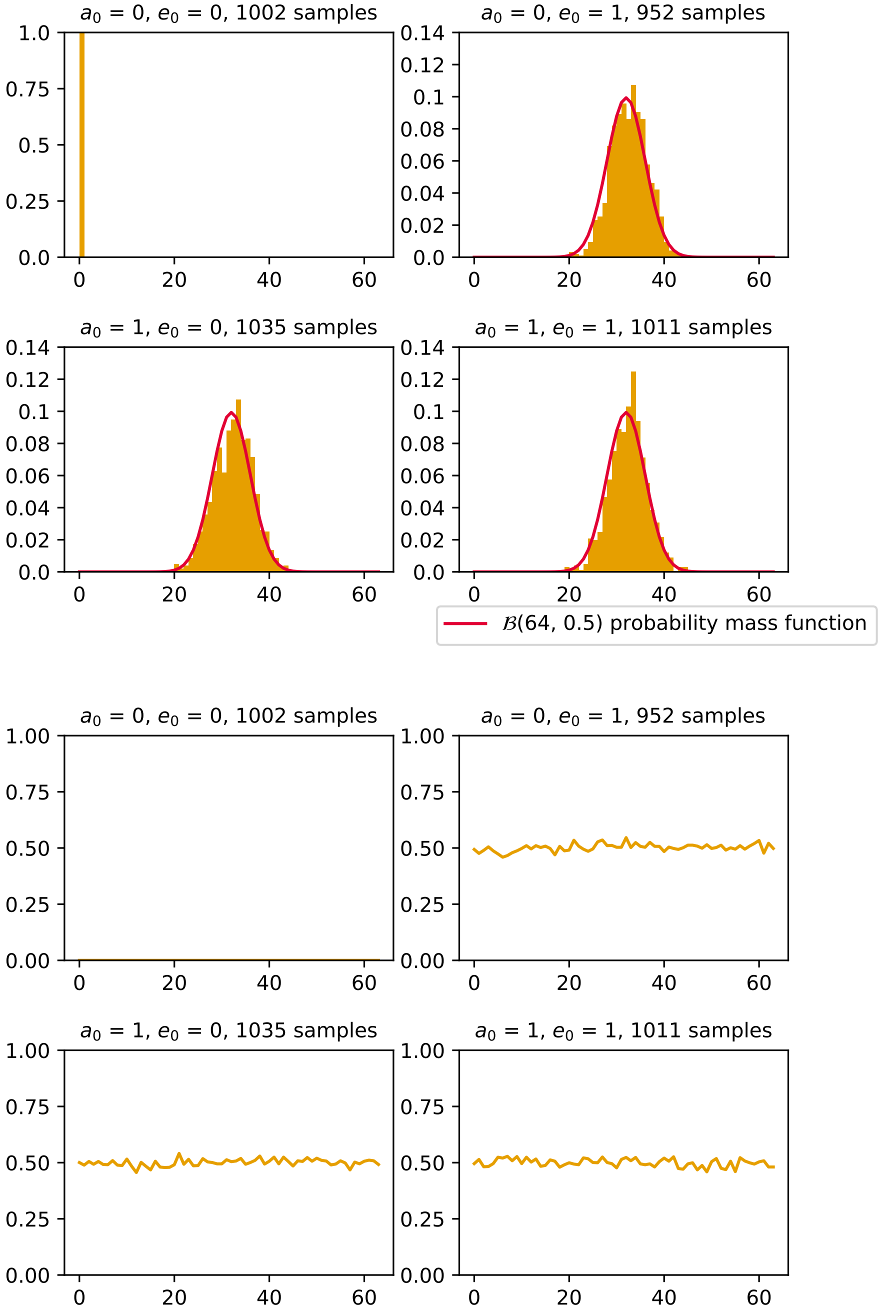


Figure 3.7: Distribution of the Hamming weight of $(\alpha^{(6)}_{w,i})_{i\in[\![0,63]\!]}$ (top) and balancedness of each $\alpha^{(6)}_{w,i}$ for any $i \in [\![0, 63]\!]$ (bottom) for 4000 random inner states sorted according to the value of (a_0, e_0).

are upper-bounded by $128 \times 2^{32} = 2^{39}$, while the memory cost is negligible. In the worst case, only e is recovered. This happens when e is the all-one vector $e = (1, \dots, 1)$. On the other hand, in the best case scenario both e and a are recovered and this happens when e is the zero vector.

Finally, note that x^v, x^w are arbitrary choices within the class $\mathcal{C}(a_0 + 1, e_0)$ and $\mathcal{C}(a_0, e_0)$. We expect that any other choice would behave in a similar manner.

3.4.4.b Second step: recovery of the remaining bits a_i

At the end of the first step, all bits e_j for $j \in [\![0, 63]\!]$ are recovered while only the bits a_j where $j \in [\![0, 63]\!] \setminus \mathrm{Supp}(e)$ are recovered. The objective of Step 2 is therefore to start from a set $U = \{j, a_j \text{ is still unknown}\} = \mathrm{Supp}(e)$ and to iteratively update its contents so that U becomes as small as possible.

To do so, we rely on the knowledge from Step 1, but also from the knowledge acquired throughout Step 2. This step therefore consists in an *adaptative* cube attack.

Adaptative choice of cubes. Let $u \in \mathbb{F}_2^{64}$ of Hamming weight 32. As shown by Table 3.3 and Corollary 3.13, any coefficient $\alpha_{u,i}^{(6)}$ depends on at most 64 variables which are a_j, e_j with $j \in \mathrm{Supp}(u)$. But because of Step 1, the 32 *values* of e_j with $j \in \mathrm{Supp}(u)$ are actually known. So only at most the 32 variables a_j with $j \in \mathrm{Supp}(u)$ remain. By choice, we can select u such that the actual number of unknowns is far less than 32. This has two consequences.

- First, by replacing variables by their known values, the ANF becomes very sparse. This happens either because terms are known to be 0, or because two terms are simplified and cancel out, for instance $e_0 a_2 a_3 x^u + e_1 a_2 a_3 x^u$, if e_0 and e_1 are known to be equal to 1. This sparsity obtained through partial knowledge enables us to compute *in practice* the ANF of $\alpha_{u,i}^{(6)}$ in which the known variables are substituted by their values. This is what the `ComputeCoefficients` procedure in Algorithm 2 performs.

- Secondly, by deliberately lowering the number of unknowns, we also lower the degree of the obtained expressions. Therefore for each choice u, we obtain 64 equations (one for each $(\alpha_{u,i}^{(6)})_{i \in [\![0,63]\!]}$) of small degrees and in a few variables. Such systems are usually easy to solve.

This is precisely the way the second step works, as described in Algorithm 2.

In our experiments, we tried to limit the number of unknowns variables to 4 or 5, that is, $|U \cap \mathrm{Supp}(u)| \in \{4, 5\}$. This might not always be possible, especially at the first iteration where $U = \mathrm{Supp}(e)$. For the first iteration, the number of variables can be limited to 5 in 91.5% of the time, and to 9 in 99.2% of the time.

Algorithm 2 Overview of Step 2

```
Input: $U = \{j, a_j$ is still unknown$\}$.
  $A, E$, sets of index-value pairs $(i, v)$ corresponding to the recovered bits $a_i$ (resp. $e_i$) during Step 1.
Output: $a_j$ for most $j$ in $U$.
  while U $\neq \emptyset$ do
    Choose $u \in \mathbb{F}_2^{64}$: wt$(u) = 32$, $U \cap \mathrm{Supp}(u) \neq \emptyset$, preferably $|U \cap \mathrm{Supp}(u)| \leq 5$.
    $P \leftarrow$ ComputeCoefficients$(u, A, E)$                    ▷ $P$, 64 polynomials.
    $\varepsilon \leftarrow$ CubeSumVector$(x^u)$
    $S \leftarrow$ SolvePolynomialSystem$(P, \varepsilon)$              ▷ $S$, set of index-value
    $A \leftarrow A \cup S$                                           pairs of recovered values.
    $U$.remove$(\{i$ such that $(i, v) \in S\})$
  end while
```

Implementation of `ComputeCoefficients`. In order to recover the ANF of $\alpha_{u,i}^{(6)}$ where the known variables are substituted by their values, we compute a partial ANF of p^r, $r \in [\![1, 6]\!]$, round after round. Because of Proposition 3.11, this partial ANF contains only highest-degree terms x^v. Furthermore, each x^v satisfies $v \preceq u$ as they are the only terms that can influence the coefficient of x^u at the sixth round. This way, the computation of a partial coefficient only took a few seconds and less than 16 Go of RAM. This can be considered as an implemented version of the *Partial Polynomial Multiplication* method introduced by Rohit *et al.*[Roh+21].

System solving. The obtained systems were also quickly solved. To do so, we used Cryptominisat [SNC09] which is a SAT-based solving algorithm. Note that linearization might also be possible as there are only $2^5 = 32$ monomials in 5 variables and we have at hand 64 equations. The choice of u, and especially the value of $|L \cap \mathrm{Supp}(u)|$ affects the cost of solving the system. The only heuristic that we used was to lower the number of unknowns as much as possible. It gave in practice good results.

Finalizing the second step. However, quite unexpectedly, the very last bits of a may be harder to recover. The problem is not the system solving, but finding an interesting system instead. Indeed, the more variables are known, the more likely we are to build a partial ANF of a coefficient which happens to be constant. In that case, there is no problem with stopping Step 2 with a few remaining unknowns a_j. As shown below, the next step requires only a few values of a_j with $j \in \mathrm{Supp}(e)$. The other ones can be recovered in the final exhaustive search. Overall, the online cost of Step 2 is less than 64 computations of derivations along spaces of dimension 32. The time complexity of building and solving the systems is harder to predict. According to our experimentation, with proper choices from the adversary, these costs remain negligible compared to the time complexity of the cube-sum computations.

3.4.4.c Third step: recovering most of the bits b_i and c_i

At this stage, we expect that all the bits e_j are recovered, as well as almost all a_j. Step 3 then consists in recovering bits b_j and c_j for all $j \in [\![0, 63]\!]$, while d_j can then be computed as $d_j = e_j \oplus c_j \oplus 1$. To do so, we cannot leverage highest-degree terms which only depend on variables e_j, a_j. We therefore mount a cube attack targeting *sub-leading monomials* x^u, that is, monomials of degree 31. As the second step, the third one is *adaptative* and is based on the computation of the expression of $\alpha_{u,i}^{(6)}$ where the recovered variables are substituted by their values.

Quadratic equations in the remaining unknowns. Because all e_j and most a_j bits have been recovered, almost all coefficients $\alpha_{u,i}^{(1)}$ where $\mathrm{wt}(u) = 1$ are recovered and can be considered constant. Corollary 3.15 can thus be greatly simplified.

Corollary 3.22. *Let* $r \in [\![2, 6]\!]$. *Let* $u \in \mathbb{F}_2^{64}$ *such that* $\deg_a(x^u) = 2^{r-1} - 1$. *Let* $i \in [\![0, 319]\!]$ *such that* $\alpha_{u,i}^{(r)} \neq 0$. *Let us assume that* a_j *and* e_j *are* known *for all* j, $j \in \mathrm{Supp}(u)$. *Then* $\alpha_{u,i}^{(r)}$ *can be expressed as a sum whose terms have one of the following forms:*

- *a binary constant, or*
- *a quadratic polynomial with monomials of the form* b_j, c_j, b_jc_j *for all* $j \in [\![0, 63]\!]$.

Proof. This is a direct consequence of Corollary 3.15 where the coefficients of all degree-1 terms are replaced by known constants. The first case of Corollary 3.15 boils down to a product of known constants, that is, to a known constant. The second case boils down to the product of a constant with a coefficient $\alpha_{0,i}$ after $p_C^{(2)}$, for some i. This coefficient is (up to an addition by 1) equal to $\alpha_{0,i}^{(1)}$. But $\alpha_{0,i}^{(1)}$ is the sum of three coefficients $\alpha_{0,\ell}^{(0.5)}$ for some ℓ. The expressions of $\alpha_{0,\ell}^{(0.5)}$ are given in Table 3.3. Only the monomials b_j, c_j, b_jc_j remain once d_j is expressed as the sum $c_j + e_j + 1$, and e_j, a_j are replaced by their known values. □

In the case where some a_j with $j \in \mathrm{Supp}(u)$ are still unknown, the same corollary holds. In that case however, the monomials a_j, a_jb_j, a_jc_j can also appear for all j such that a_j is unknown.

Any coefficient $\alpha_{u,i}^{(6)}$ with $\deg_{\mathrm{a}}(u) = 31$ can therefore be expressed as a sparse quadratic polynomial. Indeed, if $|U| \geq 0$ values of a_j remain unknown before Step 3, any $\alpha_{u,i}^{(6)}$ with $\deg_{\mathrm{a}}(u) = 31$ depends, after substitution, on at most $128 + |U|$ linear terms and $64 + 2|U|$ quadratic terms.

Avoiding useless equations. Contrary to the second step, some unwanted choices of u, where $(\alpha_{u,i}^{(6)})_{i\in[\![0,63]\!]}$ does not give information, can be avoided thanks to the following analysis.

Proposition 3.23. *Let $u \in \mathbb{F}_2^{64}$ be such that $\deg_a(x^u) = 16$. Let $i \in [\![0, 319]\!]$. Let $\prod_{j\in\mathrm{Supp}(u)} L_j \neq 0$ be a product appearing in the decomposition of $\alpha_{u,i}^{(5)}$ given by Corollary 3.13. Then, there exists $j \in \mathrm{Supp}(u)$ such that $L_j = e_j$.*

Proof. This is proved by keeping track, at round $r \in [\![1, 5]\!]$ of the coordinates $i \in [\![0, 319]\!]$ in which, any $\alpha_{u,i}^{(r)}$ (for $u \in \mathbb{F}_2^{64}$, $\mathrm{wt}(u) = 2^{r-1}$) satisfies the announced property. Bounds on the number of e_i appearing in each product are given in Table 3.6. □

Round	1	2	3	4	5	6
Row r_0	0	0	1-2	1-4	**1**-7	2-13
Row r_1	0	1	1-2	0-3	**1**-6	3-15
Row r_2	x	1	0-1	1-2	**3**-7	3-15
Row r_3	1	0-1	0-1	1-3	**2**-8	2-15
Row r_4	0	0	1	2-4	**1**-7	2-13

Table 3.6: Lower and upper bounds on the number of e_j variables appearing in each product (Corollary 3.13) of the coefficients $\alpha_{u,i}^{(r)}$, where $\deg_a(x^u) = 2^{r-1}$.

Corollary 3.24. *Let $u \in \mathbb{F}_2^{64}$ be such that $\deg_a(x^u) = 31$. Let us suppose that $e_j = 0$ for all $j \in \mathrm{Supp}(u)$. Then, for any $i \in [\![0, 319]\!]$, $\alpha_{u,i}^{(6)} = 0$.*

Proof. This is a direct consequence of Proposition 3.23, together with Corollary 3.15. □

The choice of monomials x^u for Step 3 is therefore made so that $\mathrm{Supp}(u) \cap \mathrm{Supp}(e) \neq \emptyset$.

Remark 3.25. Such choice is not possible if $e = 0$. However, in that case, it is expected that recovering b and c should be easier than in any other cases. This is highlighted by the observation made with Algorithm 1: the case $e = 0$ is the best case scenario where the entire Step 2 can be skipped. ▷

Apart from the selection process of the monomials, Step 2 and Step 3 follow the same process, already presented in Algorithm 2.

Some possible trade-offs. In that case however, the set $\mathrm{Supp}(u) \cap \mathrm{Supp}(e)$ only affects the number of variables and the sparsity of the system, but not its degree, which is bounded by 2. A small set $\mathrm{Supp}(u) \cap \mathrm{Supp}(e)$ makes the procedure `ComputeCoefficients` faster, while a larger one allows the recovery of more unknowns in one stroke. We prefer keeping $|\mathrm{Supp}(u) \cap \mathrm{Supp}(e)|$ around 4 or 5. In that case, for a given u with $\deg_{\mathrm{a}}(x^u) = 31$, computing in parallel the expressions of $(\alpha_{u,i}^{(6)})_{i \in [\![0,63]\!]}$ took about 5 to 25 minutes on two AMD EPIC 7352 (24 cores, 2.3GHz). The long computations were skipped by using the intermediate number of terms as indicator of whether or not the remaining computations will be too costly.

Overall, at least 128 equations are needed to recover the 128 bits of b and c, so at least two derivatives along spaces of dimension 31 have to be computed. Indeed, contrary to the case of coefficients of highest-degree terms, the dependency of sub-leading monomials is not only limited to variables b_j, c_j with $j \in \mathrm{Supp}(u)$. Using 3 random monomials is *often* enough to recover all the unknown bits except at most 10.

3.4.4.d Finalizing the recovery

After the three steps, a small number of unknown bits of the inner state is expected to remain. They can be recovered through an exhaustive search. In the end, the complexity of the attack is dominated by the cost of the cube-sums of degree-32 monomials, that is, by $(128 + 64)2^{32} \approx 2^{7.6+32} < 2^{40}$ both in time and data.

It is hard to give an explicit formula for the time complexity of the adaptative phases during Steps 2 and 3. However, from our experiments, they can be effectively mounted on a personal computer or a single cluster node. The entire process did not last more than a few hours on two AMD EPIC 7352.

3.4.5 Analyzing the attack with the benefit of hindsight

In this section, we comment the attack described in Section 3.4. First, we identify the properties that make our attack possible. This is particularly important from a designer's point of view. Next, we compare our attack with the independent work and Chang, Hong & Kang [CHK22] which was published around the same time as our work, and which shares similarities with it.

3.4.5.a Counter measures

The first step that is described in Section 3.4.4.a is the cornerstone of the attack. Indeed, by using conditional cubes, this step enables us to recover 96 bits in average of a and e, *without having to compute the exact ANF* of any coefficient $\alpha_{u,i}^{(6)}$. From there, the security of the remaining bits collapses with an avalanche effect. We thus consider counter-measures that avoid the use of cubes similar to those involved in Step 1.

As in all previous cube attacks against Ascon, the targeted monomials in Step 1 are chosen to have the highest possible degree. This is motivated by an already-mentioned heuristic: let D be a bound on the degree of a function F. If F is used with both public variables $x_0, \dots, x_{n-1}$ and private ones $k_0, \dots, k_{\kappa-1}$, then the coefficient $\alpha_{u,i} \in \mathbb{F}_2[k_0, \dots, k_{\kappa-1}]$ of x^u cannot be of degree more than $D - \mathrm{wt}(u)$. Targeting highest-degree term then *a priori* provides the simplest possible equations. From the analysis made in Proposition 3.11 and Corollary 3.13, this also enables us to consider 64 equations in only 128 unknowns, out of the 256 possible ones. Finally, because of the quadratic Sbox and the small number of rounds, targeting this highest-degree cube is still possible in practice.

Increasing the number of rounds. The most natural idea to prevent such an attack is to increase the number of rounds during encryption. By adding a single round, a highest-degree term would be a term of degree 64, and there exists a single such term in 64 variables. This means that only a single highest-degree term could be exploited. Each of its 64 coefficients $\alpha_{u,i}^{(7)}$ with $i \in [\![0, 64]\!]$ would depend a priori on 128 private variables, and the data limitation of 2^{64} would be reached with the single derivation along the 64-dimensional space. The major drawback is of course an increased encryption and decryption cost, as well as a lowered throughput.

Changing the outer-state row. A more-interesting counter-measure could be to change the row corresponding to the outer state. Indeed, we could consider XORing the plaintext with any of the four other Rows 1, 2, 3 or 4 instead of Row 0. This would have no effect on the performance of Ascon. However, contrary to its affine-equivalent form χ [Ber+11], the Sbox of Ascon is not rotation-invariant. Therefore, changing the position of the outer state changes the algebraic properties of the considered function, because public and private variables do not play the same role anymore. We carried out the same study as the one described in Section 3.4.2 for each of the four other possibilities and tried to find some conditional cubes in the line of Section 3.4.3. The main observations are summarized in Table 3.7. The sets mentioned in the third column of this table are similar to the ones described in Table 3.4. For instance, the first row of Table 3.7 should be understood as, for 5 indices j, the coefficients $\alpha_{u,i}^{(2)}$ where $x^u = x_0 x_j$ have $(a_0 + b_0 + d_0 + 1)$ as common linear divisor.

As a result, any of the four other choices is achieving a better resistance against our method than the current setting. Indeed, when the outer state corresponds to Row 2, the average number of recovered bits is 32, while it is 96 for Row 0. The other three scenarios do not enable us to find any conditional cube of dimension 32.

Our method leverages the slow diffusion of the public variables through the first rounds, and more specifically the low number of distinct quadratic monomials. Inserting public variables in different columns while the Sboxes are applied column-wise limits the number of quadratic monomials in public variables after the first rounds. When the outer state corresponds to Row 0, this is accentuated by the

Initial state	Linear terms after S_1	Cardinality	Analysis
a_0	$(a_0 + b_0 + d_0 + 1)x_0$	$\lvert Z_{a_0+b_0+d_0+1}\rvert = 5$	$5 + 3 + 5 + 12 < 31$ No cube as in Section 3.4.3.
x_0	$(b_0 + c_0 + 1)x_0$	$\lvert Z_{b_0+c_0+1}\rvert = 3$	
b_0	x_0		
c_0	x_0		
d_0	$(a_0 + d_0 + 1)x_0$	$\lvert Z_{a_0+d_0+1}\rvert = 5$	
		$\lvert Z_0\rvert = 12$	
a_0	$(b_0 + 1)x_0$	$\lvert Z_{b_0+1}\rvert = 4$	$4 + 6 + 23 > 31.$ Cubes can be built as in Section 3.4.3. The event $(b_i, c_i) = (1, 0)$ should be detected. Avg. recovered bits: 32.
b_0	$(b_0 + c_0 + 1)x_0$	$\lvert Z_{b_0+c_0+1}\rvert = 6$	
x_0	x_0		
c_0	x_0		
d_0	$*$		
		$\lvert Z_0\rvert = 23$	
a_0	x_0		$3 + 4 + 5 + 12 < 31$ No cube as in Section 3.4.3.
b_0	$(b_0 + c_0 + 1)x_0$	$\lvert Z_{b_0+c_0+1}\rvert = 3$	
c_0	d_0x_0	$\lvert Z_{d_0}\rvert = 4$	
x_0	$(a_0 + 1)x_0$	$\lvert Z_{a_0+1}\rvert = 5$	
d_0	x_0		
		$\lvert Z_0\rvert = 12$	
a_0	b_0x_0	$\lvert Z_{b_0}\rvert = 5$	$5 + 4 + 5 + 5 + 12 = 31$ but b_0 and $b_0 + 1$ cannot be used at the same time. No cube as in Section 3.4.3.
b_0	x_0		
c_0	$(d_0 + 1)x_0$	$\lvert Z_{d_0+1}\rvert = 4$	
d_0	$(a_0 + 1)x_0$	$\lvert Z_{a_0+1}\rvert = 5$	
x_0	$(b_0 + 1)x_0$	$\lvert Z_{b_0+1}\rvert = 5$	
		$\lvert Z_0\rvert = 12$	

Table 3.7: Overview of conditional cubes when the outer state is moved to Row 1, 2, 3 or 4.

absence of all public variables in the third row after one round. It also occurs in the only case where conditional cubes can be built, that is when the outer state corresponds to Row 2. When initialization is targeted, as done in [LDW17, Dob+15, Roh+21, RS21], the same observations can be made when inserting public variables (corresponding to the nonce) on Row 3 and keeping Row 4 all-zero, or by inputting the same variables on both Row 3 and Row 4.

It seems that the sparsity of some of the coordinates of the Sbox is the main cause: another Sbox might achieve a better resistance to (conditional) cube attacks, but probably at the cost of an increased number of gates.

3.4.5.b Comparison with the work of Chang, Hong & Kang

Another conditional cube attack against nonce-misused Ascon has been exhibited in a concurrent work by Chang, Hong & Kang [CHK22]. Both our attack and theirs were proposed independently. They both use conditional cubes to first recover the 128 bits of a and e. In particular, similarities appear between our cube x^v and their `Pattern-A` as 27 out of the 32 variables involved in the cube are identical. The objective of this section is to compare both choices. The main comparison points are presented in Table 3.8.

	Our attack		[CHK22]
Recovery of a and e	Conditional cube	$=$	Conditional cube
	$Z_{a_0} \cup Z_0$	$=$	$\{v_1, \cdots, v_{27}\}$
	primary variable	$=$	*conditional cube variable*
	$\mathrm{Supp}(w) =$ $\{0\} \cup Z_0 \cup (Z_{a_0} \setminus \{52\}) \cup Z_{e_0}$	$\approx$	`Pattern-A` $=$ $\{0\} \cup Z_0 \cup Z_{a_0} \cup \{2, 9, 12, 18\}$
	2^{39} in time and data	$\approx$	$2^{44.8}$ in time and data
Recovery of b and c	*Ad hoc* cube-like attack	$\neq$	Exhaustive search
	$\approx 2^{38}$ in time and data	$\neq$	2^{128} in time

$=, \approx$ stand for exact and partial correspondence, $\neq$ stands for no correspondence.

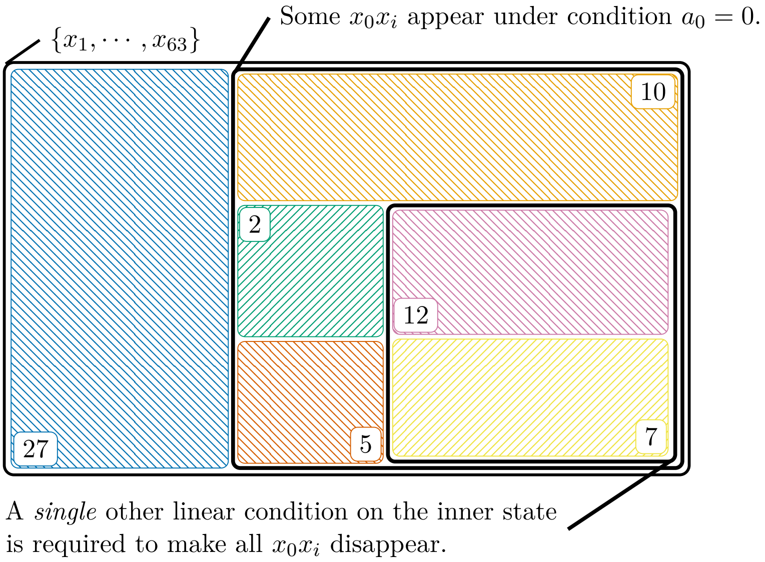
Table 3.8: Comparative study of our attack with [CHK22].

The similarities are due to a common desire to *conditionally* study the (dis)appearance of coefficients $\alpha^{(2)}_{u,i}$, where $x^u = x_0 x_j$, and which involve a_0. To do so, with x_0 as primary variable, both works start from the 27 variables from $Z_{a_0} \cup Z_0$. But 5 additional variables are required to build a monomial of degree 32. From there, the directions taken by both parties greatly differ. They can be compared by classifying the quadratic monomials in the ANF of p^2 used with private and public variables. This ANF is simple enough to be fully computed using SageMath [Sage24]. This classification is presented in Figure 3.8.

Under the assumption that $a_0 = 0$, it can be observed that 27 monomials $x_0 x_j$ do not appear: Chang, Hong & Kang selected all corresponding x_j in their pattern while we only selected 26. As explained in Section 3.4.3, this is only because in our rationale, the variables of Z_{a_0} were the last ones to be selected, and only 4 out of the 5 indices of Z_{a_0} were needed.

The remaining $63 - 27 = 36$ monomials split into subsets depending on whether or not all $x_0 x_j$ could disappear from the ANF if other conditions were added to condition $a_0 = 0$. Among them, $12 + 7 = 19$ monomials (represented in the bottom right-hand corner of Figure 3.8) can disappear from the ANF if a single linear condition (depending on j) is added to $a_0 = 0$:

- 7 monomials are such that the GCD of the coefficients of $x_0 x_i$ is divisible


Some x_0x_i appear under condition $a_0 = 0$.
$\{x_1, \cdots, x_{63}\}$
10
2
12
27
5
7
A *single* other linear condition on the inner state is required to make all x_0x_i disappear.

Variables x_i such that:

▨ no x_0x_i appear under condition $a_0 = 0$: $\mathcal{S}_a \cup \mathcal{S}_0 = \{v_1, \cdots, v_{27}\}$, see [CHK22].

Variables x_i such that some x_0x_i appear under condition $a_0 = 0$, and:

▨ no condition on the inner state will change that (some constant coefficients are present).

▨ no condition on the inner state will change that (some complementary coefficients are present).

▨ *two* linear conditions at least are needed to make them all disappear.

▨ a *single* other linear condition *different from* $e_0 = 0$ is sufficient to make them all disappear : $\{w_1, \cdots, w_{12}\}$,see [CHK22].

▨ no x_0x_i appear under conditions $(a_0 = 0, e_0 = 0)$: $\mathcal{S}_e \cup \{10, 13\}$.

Figure 3.8: Overview of the quadratic monomials $x_0x_i,\ i \in \{1, \ldots, 63\}$ after the second round.

by e_0. Their indices form the set $Z_{e_0} \cup \{10, 13\}$. We chose the remaining $32 - (1 + 26) = 5$ variables among Z_{e_0}.

- The 12 remaining ones have 12 distinct linear GCDs *independent from* e_0: they form the set $\{w_1, \cdots, w_{12}\}$ [CHK22] in which the authors chose the remaining $32 - (1 + 27) = 4$ cube variables.

Our choice corresponds to variables j such that all terms x_0x_j vanish when $e_0 = 0$ and $a_0 = 0$, see Proposition 3.17. The choice from [CHK22] is different since *a linear condition per variable* is needed in addition to condition $a_0 = 0$; hence $4 + 1 = 5$ conditions. This is the reason why their attack does not work for any value of the state Σ_{E} and needs to be repeated for 32 triples of key, nonce and associated data in average, until the state satisfies the required conditions.

Moreover, it is worth noting that minimizing the number of conditions per cube has several advantages. First of all when the vector of coefficients $(\alpha_{u,i}^{(6)})_{i \in [\![0,63]\!]}$

is not the zero vector, the adversary recovers the OR of all negated conditions. The fewer conditions, the easier it is to combine these multiple ORs to recover more bits as it is done in the last *else* case of Algorithm 1. Furthermore, the fewer conditions, the easier it is to partition all inner states into disjoint subsets, thus enabling *independent* recoveries of bits, as shown by the independent use of all $x^{v \lll j}$ for $j \in [\![0, 63]\!]$.

The recoveries of the last 128 bits are also entirely different in both works, as pointed out in Table 3.8.

3.5 Concluding remarks

As shown in this chapter, higher-order differential techniques are redoubtable vectors of attacks against block ciphers. They can leverage the inherent structure of the overall construction of a round function, as shown with the Square attack, but they also benefit from the simplicity of the algebraic normal form. Such flaws can be detected using automatic tools or other methods related to the division property, but also by hand. In that case, a careful analysis of the successive ANF of the first rounds can lead to attacks as the one presented in Section 3.4. However, such analyzes are only reserved to ciphers where the ANF is sufficiently sparse and structured. They moreover highly rely on the small growth of degree. This is the main reasons why higher-order differential attacks can often only be applied to round-reduced versions of cryptographic primitives, or to primitives based on low-degree round functions.

Today, while very-optimized tool-based methods lead to relatively small gain, the attacks mounted by hand are not in the spotlight anymore and have trouble taking on new forms. This is however the latter ones that nourish the former ones. In the following, we therefore expose some directions that, to the best of our knowledge, have never been investigated, and which seem, at least on paper, interesting to pursue.

Data-optimized cube attacks. As mentioned many times, most, if not all, cube attacks target monomials of highest-degree, due to the *a priori* simplest form of their coefficients. This implies a high cost in data as derivation along $\mathrm{Prec}(u)$ requires the value of $F(v)$ for any $v \preceq u$. This high data requirement is in practice *never* exploited to its fullest. Indeed, the knowledge of $\{F(v), v \preceq u\}$, not only enables the attacker to recover the value of the coefficient of x^u (see Proposition 2.10 or Eq. (3.1)), as the $2^{\mathrm{wt}(u)}$ coefficients of monomials x^v with $v \preceq u$ can be recovered as well. These coefficients are *a priori* more intricate, but could be made sparser after the recovery of some first bits.

Conditional distinguishers. A related idea is to directly focus on coefficients of low degree. Again, their *a priori* complexities is counter-balanced by the low data cost of evaluating them. This therefore enables us to target a lot of them, rather than just a few costly ones. It is then possible to accept the fact of recovering information only one out of many times: this may happen for instance for coefficients $(\alpha_{u,i})_{i\in[\![0,63]\!]}$ with very intricate polynomial representations, which degenerate into very simple polynomials under some conditions on the unknowns. This is for example the case of the coefficients targeted in Section 3.4.3, which, under two linear conditions, collapse into the zero polynomial. Such coefficients could be used in attacks, especially if querying their values is not too costly.

This direction is in particular made possible by the efficient and exact techniques mentioned in Section 3.1.2.a that can compute the exact ANF of some coefficients. In most of the cases such coefficients cannot be directly used in an attack because they are too dense. But if a heuristic method was able, given a *known* polynomial to effectively point out conditions under which this polynomial is simplified, for instance becomes unbalanced, this would be highly beneficial to the field.

Probabilistic distinguishers. Finally, while the recent work of Hu, Peyrin, Tan & Yap [Hu+23] is very refreshing and proved to be very powerful, it seems that their framework could be clarified. In the light of Example 3.7, the so-called *higher-order algebraic transitional form* and the *differential supporting function* are not yet clearly understood (at least to the author of this thesis) and could benefit from further analysis. As shown by the case of derivatives of order 1, probabilistic higher-order differential distiguishers should in practice be way more versatile and powerful than the more restrictive deterministic ones.

Chapter 4

From invariants to the differential cryptanalysis of conjugates

While the exact theoretical definition of lightweight cryptography is hard to grasp [BP17], determining the different trends in this field may be a way to better understand it. One of the main criteria that are taken into account by designers of lightweight primitives is the hardware area, that is, the size of the implemented circuit. More recently, as highlighted for instance by the block ciphers Prince [Bor+12], Mantis [Bei+16], or QARMAv2 [Ava+23] achieving a low-latency started to gain more and more attention. The designers of Midori [Ban+15] however took another approach by trying to reduce the energy consumption of a cipher. To do so, in the original paper [Ban+15], they analyze all the components of their primitive under construction and present a cipher that is heavily inspired by the AES, together with a thorough initial cryptanalysis.

In the following years, this cipher attracted many third-party analyses [GL16, TLS16, Guo+16, LW17, BCC19, Bey18, BCL18, TLS19, Bey21]. In particular, the invariant subspace attack by Guo, Jean, Nikolić, Qiao, Sasaki & Sim [Guo+16], the non-linear invariants by Todo, Leander & Sasaki [TLS19] or the invariants by Beyne [Bey18] all enable us to distinguish Midori (or slightly-modified versions of it) in a weak-key setting. Even if all of those attacks point out that the choices of the designers were too aggressive, it is still unclear what exactly goes wrong. This is actually not an isolated case: many other designs such as iSCREAM [LMR15], NORX v2.0 [Cha+17], Simpira v1 [Røn16] or Haraka v1 [Jea16] are also threatened by such attacks.

In this chapter, our first goal is then to draw a precise description of Midori and a review of the aforementioned cryptanalyses. In particular, the close relationships between them are exhibited. In a second phase, we continue the analysis effort of Midori by presenting a differential cryptanalysis, not of Midori directly, but of one of its conjugates. This technique is highly-inspired from the non-linear conjugate framework by Beierle, Canteaut & Leander [BCL18]. As in the latter work, this technique shows that an adversary can always choose to analyze a cipher in a system of variables in which differential or linear flaws are easier to detect and stronger than expected, at least in a weak-key setting. Afterwards, we translate the differential properties of conjugates $G \circ F \circ G^{-1}$ of a cryptographic function F into *inherent* properties of F, which points out *commutative cryptanalysis* as a natural generalization of standard differential cryptanalysis. Finally, we bridge the gap between differential cryptanalysis of conjugates, commutative cryptanalysis and, differential cryptanalysis using alternative group laws that is introduced by Civino, Blondeau & Sala [CBS19]. Hopefully, the obtained dictionary will help understanding the underlying security notions that a designer should target.

This chapter is based on a joint work with Patrick Felke, Gregor Leander, Patrick Neumann, Léo Perrin & Lukas Stennes that is published at IACR Transactions on Symmetric Cryptology, 2022(4) [Bau+23], and on an on-going work with the same coauthors, together with Christof Beierle. While the differential cryptanalysis of conjugates is briefly mentioned in the appendix of the published paper, this technique is described in detail in this chapter.

Contents

4.1 Description of Midori

4.1.1 Context around Midori

Midori is a family of block ciphers designed by Banik, Bogdanov, Isobe, Shibutani, Hiwatari, Akishita & Regazzoni [Ban+15] in 2015. This primitive is designed to be competitive with respect to energy consumption, but also to make decryption available with no significant area or energy overhead. This latter property is also sought by the reflection ciphers Prince [Bor+12] or Mantis [Bei+16]. As described in the following sections, both criteria heavily influenced the design of the family.

The Midori family is composed of two members, namely Midori64 which has a state of 64 bits, and, Midori128 which has a state of 128 bits.

Both Midori64 and Midori128 are ciphers using a 128-bit key. The two designs are AES-like ciphers that are very similar one to the other. This is the reason why we first present in Section 4.1.2 the 64-bit-state version which is the most widely analyzed of both, and in particular by us in Sections 4.3 and 5.1. Based on its sibling, Midori128 is then succinctly described in Section 4.1.3. At the same time, we take the opportunity to present and give a name to variants of both Midori64 and Midori128, that were already analyzed in previous works, and that we further study in Section 5.4.1.a.

4.1.2 Midori64

As in the AES (see Example 1.7), the 64-bit state of Midori64 is viewed as a 4×4 matrix, but this time of 4-bit nibbles. These nibbles are sometimes called *cells* in the following. The cells are numbered from top to bottom, and from left to right, as depicted in Figure 4.1, and the enumeration is done from the least significant nibble to the most significant one. For instance, the nibble of index 0 (resp. 12) of the word `0xfedcba9876543210` is equal to `0x0` (resp. `0xc`), and is located at the top left-hand (resp. right-hand) corner of the matrix representation.

The round function is therefore built with this matrix representation in mind, as shown on Figure 4.2. The different layers of the round function are described in the following section.

0	4	8	12
1	5	9	13
2	6	10	14
3	7	11	15

Figure 4.1: The 64-bit state of Midori64 seen as a 4×4 matrix of 4-bit nibbles.

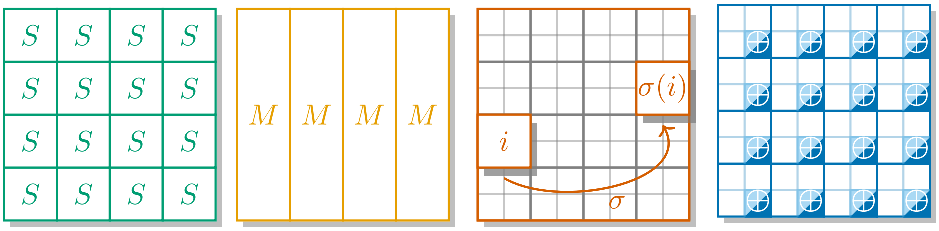


Figure 4.2: The Sbox layer, MixColumns, ShuffleCells, and constant addition layer of Midori64.

4.1.2.a The round function

The round function of Midori64 follows a standard SPN construction which alternates between linear and non-linear layers.

The Sbox layer. The first operation is the Sbox layer. This layer is built as the parallel application of a single 4-bit Sbox $S\colon \mathbb{F}_2^4 \to \mathbb{F}_2^4$. This Sbox S is therefore applied to each nibble. We denote by $\mathcal{S}$ the Sbox layer, so that $\mathcal{S}\colon (\mathbb{F}_2^4)^{16} \to (\mathbb{F}_2^4)^{16}$ is defined by:

$$\mathcal{S}\colon (x_0, \ldots, x_{15}) \mapsto (S(x_0), \ldots, S(x_{15})).$$

The look-up table of the Sbox S is given in Table 4.1.

x	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
$S(x)$	c	a	d	3	e	b	f	7	8	9	1	5	0	2	4	6

Table 4.1: Look-up table of the Sbox of Midori64 in hexadecimal notation.

We non-exhaustively list some of its noteworthy properties.

- It is an involutive Sbox, *i.e.* S satisfies $S \circ S = \mathrm{Id}$. In other words, S is bijective and its inverse is S itself.
- The linearity of S is $\mathcal{L}(S) = 8$.
- The differential uniformity of S is $\delta_S = 4$.
- The Sbox S has four fixed points: $S(\texttt{0x3}) = \texttt{0x3}$, $S(\texttt{0x7}) = \texttt{0x7}$, $S(\texttt{0x8}) = \texttt{0x8}$ and $S(\texttt{0x9}) = \texttt{0x9}$.

The MixColumns layer. Then, a MixColumns operation is applied. This operation, that we denote by $\mathsf{MC}\colon \mathbb{F}_2^{64} \to \mathbb{F}_2^{64}$, is defined as the parallel application of a single $\mathbb{F}_{2^4}$-linear operation $M\colon (\mathbb{F}_{2^4})^4 \to (\mathbb{F}_{2^4})^4$ whose matrix $M \in \mathbf{M}_4(\mathbb{F}_{2^4})$ is defined by:

$$M := \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}. \tag{4.1}$$

Contrary to the matrix of the MixColumns operation in the AES, this matrix is not MDS, but almost MDS as its branch number is equal to 4. While diffusion is not optimal, the advantage is of course that the MixColumns of Midori is way cheaper to implement than the one in the AES. Indeed, while M belongs to $\mathbf{M}_4(\mathbb{F}_{2^4})$, we can further observe that it is only made of copies and addition: given a column, the i-th output cell is the sum of the three input cells of index different from i. As the addition in $\mathbb{F}_{2^4}$ coincides with the bitwise addition in $\mathbb{F}_2^4$, this further implies that the MixColumns operation actually corresponds to 16 parallel applications of $N\colon (\mathbb{F}_2)^4 \to (\mathbb{F}_2)^4$, where N is defined "just as M":

$$N := (x_0, x_1, x_2, x_3) \mapsto (x_1 + x_2 + x_3, x_0 + x_2 + x_3, x_0 + x_1 + x_3, x_0 + x_1 + x_2).$$

In other words, it should be noted that M provides no intra-nibble mixing: given a column, and $i, j \in [\![0, 3]\!]$, the j-th bit of the i-th output cell only depends on the j-th bits of the three input cells with index different from i. The function M is also involutive; the full MixColumns layer MC therefore inherits from the same property.

The ShuffleCells layer. Afterwards, the cells of the state are reorganized during the ShuffleCells operation, that is denoted by SC. More precisely, there exists a permutation $\sigma\colon [\![0, 15]\!] \xrightarrow{\sim} [\![0, 15]\!]$ such that the i-th output cell after SC, exactly corresponds to the $\sigma(i)$-th input cell before SC. The look-up table of σ is given in Table 4.2, and a graphical representation of SC is given in Figure 4.3.

x	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$\sigma(x)$	0	7	14	9	5	2	11	12	15	8	1	6	10	13	4	3

Table 4.2: Look-up table of the permutation of cells σ.

It can be observed that the ShuffleCells operation is noticeably less structured than the ShiftRows operation of the AES.

`0x0`	`0x4`	`0x8`	`0xc`
`0x1`	`0x5`	`0x9`	`0xd`
`0x2`	`0x6`	`0xa`	`0xe`
`0x3`	`0x7`	`0xb`	`0xf`

`0x0`	`0x5`	`0xf`	`0xa`
`0x7`	`0x2`	`0x8`	`0xd`
`0xe`	`0xb`	`0x1`	`0x4`
`0x9`	`0xc`	`0x6`	`0x3`

Figure 4.3: Matrix representation of the vectors `0xfedcba9876543210` (left) and SC(`0xfedcba9876543210`) (right).

The constant and key addition layer. Finally, additions of constant and round key occur. Given a round index $r \in [\![0, R-2]\!]$, where $R \geq 1$ is the number of rounds, we denote by $c^{(r)}$ and $k^{(r)}$ the respective 64-bit round constant and 64-bit round key added at the end of the round. As $r \leq R-2$, no addition layer takes place in the last round, but the last round is followed by a post-whitening. The constants used in Midori are very sparse: if $c^{(r)}$ is seen as a 16-nibble-long vector, then $c^{(r)} \in \{0,1\}^{16}$. In other words, the constant addition only affects the least significant bit of each nibble of the state.

The whole round function. Finally, given a round index $r \in [\![0, R-1]\!]$, we denote by $F_k^{(r)} \colon \mathbb{F}_2^{64} \to \mathbb{F}_2^{64}$ the r-th round function. If $r \in [\![0, R-2]\!]$, $F_k^{(r)}$ is therefore defined by:

$$F_k^{(r)} := T_{k^{(r)}+c^{(r)}} \circ \mathsf{SC} \circ \mathsf{MC} \circ \mathcal{S}.$$

The last round function $F_k^{(R-1)}$ only consists in an Sbox layer: $F_k^{(R-1)} := \mathcal{S}$.

4.1.2.b The full encryption

From the round functions described above, the full block cipher $\mathcal{E} = (E_k \colon \mathbb{F}_2^{64} \xrightarrow{\sim} \mathbb{F}_2^{64})$ is defined as the composition of $R = 16$ rounds, with a pre- and post-whitening. Stated otherwise, let $k \in \mathbb{F}_2^{128}$ be the master key and $\mathrm{WK} \in \mathbb{F}_2^{64}$ be the whitening key derived from it. Then E_k is defined by:

$$E_k := T_{\mathrm{WK}} \circ F_k^{(15)} \circ F_k^{(14)} \circ \ldots \circ F_k^{(1)} \circ F_k^{(0)} \circ T_{\mathrm{WK}}.$$

The key schedule. The only remaining component to describe is the generation of the whitening key WK and the round keys $k^{(r)}$ for $r \in [\![0, 14]\!]$, that are all derived from the master key $k \in \mathbb{F}_2^{128}$. This scheduling is almost trivial and (therefore) very efficient: the master key k is decomposed as $k = (k_0, k_1)$ where $k_0 \in \mathbb{F}_2^{64}$ (resp. $k_1 \in \mathbb{F}_2^{64}$) is the least (resp. most) significant half of k. Then WK is defined as the sum of k_0 and k_1, while $k^{(r)}$ is equal to either k_0 or k_1 depending on the value of r modulo 2, *i.e.*:

$$\mathrm{WK} := k_0 + k_1, \quad \text{and} \quad \forall\, r \in [\![0, 14]\!]\,,\ k^{(r)} := k_{r \bmod 2}.$$

4.1.2.c The Vert family of toy ciphers

In the following, we study not only the original Midori64, but also variants of this cipher where the permutation of cells and/or the round constants are modified.

We therefore introduce a family of toy ciphers called Vert.[1] Each instance is parameterized by a pair (ρ, d) and denoted by Vert_{ρ}^{d}. More precisely:

- $\rho \colon [\![0, 15]\!] \xrightarrow{\sim} [\![0, 15]\!]$ is a bijection defining a permutation of cells. The permutation σ defined in Section 4.1.2.a is replaced by ρ in that case.
- $d \in \mathbb{F}_2^4$ is a 4-bit constant. In that case, each round constant $c^{(r)} = \left(c_0^{(r)}, \dots, c_{15}^{(r)}\right)$ is replaced by $d^{(r)} = \left(d_0^{(r)}, \dots, d_{15}^{(r)}\right)$ where for any i, $d_i^{(r)} = d$ if $c_i^{(r)} = 1$, and $d_i^{(r)} = 0$ otherwise.

Any member of the Vert family is then a 64-bit-state and 128-bit-key cipher. We denote by $\mathsf{Vert}_{\mathsf{SC}}$ the subfamily of ciphers which uses the original ShuffleCells permutation used in Midori64, and by $\mathsf{Vert}_{\mathsf{SR}}$ the subfamily which rather uses ShiftRows, that is, the permutation of cells that is used in the AES.

In other words, if the round constants are considered to be part of the key schedule algorithms, that is, if $k^{(r)}$ is defined by $k^{(r)} := k_{r \bmod 2} + c^{(r)}$ for any r, then any instance of $\mathsf{Vert}_{\mathsf{SC}}$ can be considered as the original Midori used with a very-close key schedule. In particular, $\mathsf{Vert}_{\mathsf{SC}}^{1}$ coincides with Midori64.

Such modified versions of Midori64 have already been studied. For instance the subfamily $\mathsf{Vert}_{\mathsf{SR}}$ is considered in the original Midori paper [Ban+15], where bounds on the number of differential active S-boxes are given. In third-party cryptanalysis papers, the ciphers $\mathsf{Vert}_{\mathsf{SC}}^{d}$, with $d \in \langle 2, 8 \rangle$ have been analyzed by Beyne [Bey18, Bey20], while $\mathsf{Vert}_{\mathsf{SC}}^{5}$ is studied by Todo, Leander & Sasaki in [TLS19].

4.1.3 Midori128

The general construction. As already mentioned, the structure of Midori128 is really close to the structure of the AES, but even more to the structure of Midori64. Its 128-bit state is seen as a 4×4 matrix of bytes, that are sometimes named cells as well. The round function is also built as the composition of an Sbox layer, a MixColumns layer, a ShuffleCells layer, and a key/constant addition layer.

The MixColumns $\mathsf{MC} \colon (\mathbb{F}_{2^8})^{16} \to (\mathbb{F}_{2^8})^{16}$ is defined using the same matrix M that is defined in Eq. (4.1) but M is this time seen as a matrix of $\mathbf{M}_4(\mathbb{F}_{2^8})$, and the ShuffleCells uses the same permutation of cells σ that is given in Table 4.2.

The number of rounds R is equal to 20. The first 15 rounds constants $C^{(r)} \in (\mathbb{F}_2^8)^{(16)}$ are almost identical to the ones used in Midori64: if each byte $C_i^{(r)}$ is considered as an element of $\mathbb{F}_2^4 \times \mathbb{F}_2^4$, then $C_i^{(r)} = (c_i^{(r)}, 0)$. Therefore if $C^{(r)}$ is seen as 16-byte vector, then $C^{(r)} \in \{0, 1\}^{16}$. In other words, the first 15 round constants are the ones of Midori64, but seen as bytes instead of nibbles. The four constants

[1] As *midori* means green in Japanese, we choose the French translation of this color to name this family.

$C^{(15)}, C^{(16)}, C^{(17)}, C^{(18)}$ also belong to $\{0,1\}^{16}$. Each round key $k^{(r)}$ is equal to the master key k. This is also the case of WK.

The Sbox layer in detail. Finally, only the Sbox layer is built differently. In Midori128 four different 8-bit Sboxes are used in parallel. They are denoted by $\mathrm{SSb}_0, \mathrm{SSb}_1, \mathrm{SSb}_2, \mathrm{SSb}_3$. The first one, SSb_0, is applied four times in parallel on the first row, the second one, SSb_1, is applied four times in parallel on the second row, and so forth. The four Sboxes are depicted in Figure 4.4. As we can see on this figure, for each $i \in [\![0,3]\!]$, the Sbox SSb_i is defined by $L_i \circ \mathrm{SS} \circ L_i^{-1}$ where $L_i \colon \mathbb{F}_2^8 \to \mathbb{F}_2^8$ is a bit permutation, and $\mathrm{SS} \colon \mathbb{F}_2^8 \to \mathbb{F}_2^8$ is the application of two involutive 4-bit Sboxes in parallel. In other words, each of the Sbox SSb_i is a (linear) conjugate of SS, and therefore is involutive as well.

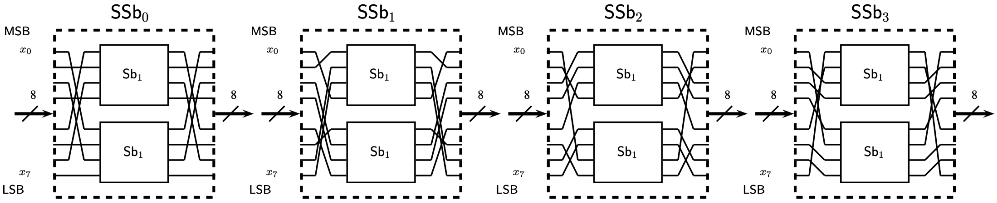


Figure 4.4: The Sboxes used in Midori128, extracted from [Ban+15].

The Grün family of toy ciphers. Finally, we define a family of toy ciphers containing Midori128 as a member. We name this family Grün in honor of the German coauthors of the paper on which this chapter is based on. This family is again parameterized by a permutation of cells used in place of SC, and by a constant $D \in \mathbb{F}_2^8$, so that the the i-th byte of the r-th round constant $D_i^{(r)}$ is equal to D if the genuine $C_i^{(r)}$ is equal to 1. Otherwise $D_i^{(r)} = 0$.

From now on, *if not explicitly mentioned*, Midori stands for Midori64 and *not* Midori128, which is only studied in Section 5.4.1.b.

4.2 Previous cryptanalyses of Midori64

With this notation and description in mind, we look back to the main third-party cryptanalyses of Midori64.

4.2.1 Invariant subspaces

Main principle. In a paper from 2016, Guo, Jean, Nikolić, Qiao, Sasaki & Sim [Guo+16] present a distinguishing attack against Midori64, when the cipher is used with very specific keys. With the naming already introduced in Section 3.3, this is thus a weak-key attack. The distinguishing property leveraged by the authors is the existence of an *invariant subspace* for the round function $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ without key/constant addition, that is, an affine space $a+V$ satisfying $F(a+V) = b+V$ for some $a, b \in \mathbb{F}_2^n$.

Let us suppose that such $a, b \in \mathbb{F}_2^n$ and $V \subset \mathbb{F}_2^n$ exist. In that case, given a round key $k^{(r)}$ in the affine space $a+b+V$, we observe that:

$$T_{k^{(r)}} \circ F(a+V) = T_{k^{(r)}}(b+V) = a+V,$$

where the last equality is obtained because for any $v, w \in V$, $(b+v)+a+b+w = a+(v+w) \in a+V$. In particular, if *all round keys* belong to the space $a+b+V$, then, no matter the number of rounds, the ciphertext corresponding to a plaintext $x \in a+V$ necessarily belongs to $a+V$. This therefore allows to distinguish a cipher used with such a weak-key from a random bijection, by using only a single chosen plaintext. This technique was introduced by Leander, Abdelraheem, AlKhzaimi & Zenner in 2011 [Lea+11], and an automatic tool to detect such invariant spaces was proposed by Leander, Minaud & Rønjom [LMR15] in 2015.

Remark 4.1. This tool not only led to the discovery of flaws in block ciphers such as iSCREAM in the original work [LMR15], it was also a cornerstone in the discovery of the alternative representation of the key schedule of AES by Leurent & Pernot [LP21]. This work indeed starts from a partition of the key space into four subspaces, which are invariant, not over one, but over four rounds of the key schedule. ▷

Invariants subspaces of Midori64. In the case of Midori64, the authors of [Guo+16] detect such an invariant by a careful analysis starting from the Sbox S.

As observed in Table 4.1 and as mentioned in Section 4.1.2.a, 0x8 and 0x9 are two fixed points of the Sbox. This means that the space $\{\texttt{0x8}, \texttt{0x9}\}^{16} = (\texttt{0x8}+V)^{16}$ where $V = \{0,1\}$ is not only an invariant space of $\mathcal{S}$, it is actually sent onto itself by $\mathcal{S}$ as $\mathcal{S}((\texttt{0x8}+V)^{16}) = (\texttt{0x8}+V)^{16}$. We denote this set by $\mathcal{V} := (\texttt{0x8}+V)^{16}$. The subspace $\mathcal{V}$ is also preserved by the ShuffleCells operation which only reorganizes the cells, and by the MixColumns layer. Indeed, if $u, v, w \in \{0,1\}$, then we observe that:

$$\texttt{0x8} + u + \texttt{0x8} + v + \texttt{0x8} + w = \texttt{0x8} + (u+v+w) \in \texttt{0x8} + \{0,1\}\,.$$

We prove in the same way the more general fact that $M((a+W)^4) = (a+W)^4$ for any linear subspace $W \subset \mathbb{F}_2^4$ and any constant $a \in \mathbb{F}_2^4$. Finally, for any r, the round constant $c^{(r)}$ belongs to V. This in particular means that:

$$T_{c_i^{(r)}}(\texttt{0x8}+V) = \texttt{0x8} + (c_i^{(r)} + V) = \texttt{0x8} + V,$$

and $\mathcal{V}$ is also preserved by any round-constant addition $T_{c^{(r)}}$. From there, we deduce that $\mathcal{V}$ is preserved by $T_{c^{(r)}} \circ \mathsf{SC} \circ \mathsf{MC} \circ \mathcal{S}$. If each round key belongs to V^{16}, then the round-key addition behaves as the round-constant addition, and an invariant subspace for the whole cipher is deduced. Because of the very light key-schedule, we observe that this is the case if the two halves $k_0, k_1 \in \mathbb{F}_2^{64}$ of k both belong to V^{16}. This is how the weak-key space of size $2^{16} \times 2^{16} = 2^{32}$ is deduced in [Guo+16]. This distinguishing property can immediately be generalized as a property of any member of Vert^1, as the permutation of cells has no influence here.

A very specific case. In this very specific case, the distinguisher is not just a standard invariant subspace: the restriction of Midori64 to plaintexts within $\mathcal{V}$ and to keys within $\{0,1\}^{32}$ is actually affine. Indeed, the only non-linear operation S becomes the identity map, and in particular a linear map, when its inputs are restricted to its fixed points. The hypothesis made on the keys enables each of the Sbox layers to satisfy this property. This is the reason why the distinguishing attack of [Guo+16] leads to a powerful key recovery in the weak-key setting that is based on solving the linear system obtained.

4.2.2 Non-linear invariants

Main principle. In the following years, Todo, Leander & Sasaki [TLS19] continued studying weak-key distinguishers. In their paper [TLS19], they again leverage the fact that, key schedules as light as the one of Midori64 allow to easily convert conditions on round keys into actual conditions on the master key.

A non-linear invariant of a block cipher $\mathcal{E} = (E_k \colon \mathbb{F}_2^n \xrightarrow{\sim} \mathbb{F}_2^n)$ is a non-linear Boolean function $f \colon \mathbb{F}_2^n \to \mathbb{F}_2$ such that for many $k \in \mathbb{F}_2^\kappa$, there exists $\varepsilon_k \in \{0,1\}$ such that Eq. (4.2) is satisfied:

$$\forall x \in \mathbb{F}_2^n, \quad f(x) + f(E_k(x)) = \varepsilon_k. \tag{4.2}$$

If such a function f is balanced, this property enables to distinguish the cipher instantiated with a weak key from a random function $F \colon \mathbb{F}_2^n \to \mathbb{F}_2$, in a *known plaintext* scenario. Indeed, the equality $f(x) + f(F(x)) = \varepsilon$ holds in average for half of the inputs $x \in \mathbb{F}_2^n$. If f is not balanced, such a distinguisher can still be mounted, only with a smaller advantage.

More generally, if there exists $f, g \colon \mathbb{F}_2^n \to \mathbb{F}_2$ and $F \colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ such that $g \circ F = f$, we denote it in the following by $f \xrightarrow{F} g$.

Example 4.2 (Invariant subspaces and non-linear invariants)**.** An invariant subspace $a + V$ that is *mapped onto itself* by a block cipher can be interpreted as a non-linear invariant. Indeed, the function $f = \mathbf{1}_{a+V}$ satisfies in that case $f \xrightarrow{E_k} f$. Furthermore, f is unbalanced unless $\dim(V) = n - 1$. This unbalancedness is compensated by considering a chosen plaintext scenario instead of a known plaintext one. If $\dim(V)$ is instead equal to $n - 1$, as f is the indicator function of an affine hyperplane, f is not only balanced, it is actually an affine function.

On the other hand, a balanced non-linear invariant can be thought of as a subset $Z \subset \mathbb{F}_2^n$ (and not a *subspace*) that is mapped onto itself (or onto its complement $\mathbb{F}_2^n \setminus Z$) by a block cipher. Indeed, any function f can be considered as the indicator function of its support:[2] $f = \mathbf{1}_{\mathrm{Supp}(f)}$ and Eq. (4.2) is equivalent to $E_k(\mathrm{Supp}(f)) = \mathrm{Supp}(f)$ or $E_k(\mathrm{Supp}(f)) = \mathbb{F}_2^n \setminus \mathrm{Supp}(f)$ depending on the value of ε_k. ▷

The restriction to non-linear Boolean functions f is not necessary. However, the existence of a linear function f satisfying Eq. (4.2) is not expected, as most of the ciphers today are built with resistance against linear cryptanalysis in mind. Indeed, in the linear case, Eq. (4.2) is equivalent to Eq. (2.16) with $\alpha = \beta$, because of Proposition 2.49. In other words, invariant subspaces generalize linear approximations holding with probability one, which correspond to indicator functions of hyperplanes, while non-linear invariants generalize the former ones to any indicator function.

Equivalent definition. By looking at Eq. (4.2), we observe that a (non-linear) invariant for E_k is nothing more than a function f such that for any cycle $\mathcal{C} \subset \mathbb{F}_2^n$ of E_k, f is either constant over $\mathcal{C}$ when $\varepsilon = 0$, or alternating between 0 and 1 over $\mathcal{C}$ when $\varepsilon = 1$. The latter case can only occur when the lengths of all cycles in the decomposition of E_k are even. In the case where some cycles have odd lengths, the set $\{\mathbf{1}_{\mathcal{C}_i}, i \in [\![0, \ell - 1]\!]\}$ of indicator functions of all cycles immediately yields a basis of the space of (non-linear) invariants. This is applicable for instance to the Sbox S of Midori64 which has fixed points and therefore cycles of length 1. We obviously deduce that invariants then exist for any function. The challenge is then to find one that is common to many E_k with different keys.

However, for a given k, the cycle decomposition of an instantiated cipher E_k is usually unavailable due to the cardinality of its domain. This is the reason why, such invariants are usually found by leveraging both the iterative construction and the simplicity of the layers.

Non-linear invariants of each layer of Midori64. For the Sbox S of Midori64, the invariants can be exhaustively listed thanks to the previous observation. The functions $f, g \colon \mathbb{F}_2^4 \to \mathbb{F}_2$ that are defined by:

$$f \colon (x_0, x_1, x_2, x_3) \mapsto x_0 + x_3 + x_0x_3 + x_2x_3, \quad g \colon (x_0, x_1, x_2, x_3) \mapsto x_0 + x_1 + x_2 + x_2x_3$$

are among this list and, as shown in Tables 4.3 and 4.4, they satisfy:

$$f \xrightarrow{S} f, \quad \text{and} \quad g \xrightarrow{S} g.$$

We naturally deduce that the functions $\widetilde{f}, \widetilde{g} \colon (\mathbb{F}_2^4)^{16} \to \mathbb{F}_2$ defined by:

$$\widetilde{f} \colon (x_0, \ldots, x_{15}) \mapsto \sum_{i=0}^{15} f(x_i), \quad \text{and} \quad \widetilde{g} \colon (x_0, \ldots, x_{15}) \mapsto \sum_{i=0}^{15} g(x_i),$$

[2] The support of a Boolean function f is defined as the support of its truth table viewed as a vector: $(f(x))_{x \in \mathbb{F}_2^n} \in \mathbb{F}_2^n$.

x	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
$f(x)$	0	1	0	1	0	1	0	1	1	1	1	1	0	0	0	0
$S(x)$	c	a	d	3	e	b	f	7	8	9	1	5	0	2	4	6
$f(S(x))$	0	1	0	1	0	1	0	1	1	1	1	1	0	0	0	0

Table 4.3: The quadratic invariant f of the Sbox of Midori64.

x	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
$g(x)$	0	1	1	0	1	0	0	1	0	1	1	0	0	1	1	0
$S(x)$	c	a	d	3	e	b	f	7	8	9	1	5	0	2	4	6
$g(S(x))$	0	1	1	0	1	0	0	1	0	1	1	0	0	1	1	0

Table 4.4: The quadratic invariant g of the Sbox of Midori64.

are invariants of $\mathcal{S}$ that satisfy:

$$\widetilde{f} \xrightarrow{\mathcal{S}} \widetilde{f} \quad \text{and} \quad \widetilde{g} \xrightarrow{\mathcal{S}} \widetilde{g}.$$

The functions $\widetilde{f}, \widetilde{g}$ are also obvious invariants of the ShuffleCells operation, which only reorganizes the cells, and therefore the sum in the previous definitions of $\widetilde{f}, \widetilde{g}$. Furthermore, let $(x_0, x_1, x_2, x_3) \in (\mathbb{F}_2^4)^4$ and $\Sigma_{\text{col}} := \sum_{i=0}^{3} g(M_i(x_0, x_1, x_2, x_3))$. We observe that:

$$\begin{aligned}\Sigma_{\text{col}} &:= \sum_{i=0}^{3} g\left(M_i(x_0, x_1, x_2, x_3)\right) = \sum_{i=0}^{3} g(\sum_{\substack{j=0,\\ j\neq i}}^{3} x_i) \\ &= \sum_{i=0}^{3} \left(\sum_{\substack{j=0,\\ j\neq i}}^{3} (x_{j,0} + x_{j,1} + x_{j,2}) + (\sum_{\substack{j=0,\\ j\neq i}}^{3} x_{j,2})(\sum_{\substack{\ell=0,\\ \ell\neq i}}^{3} x_{\ell,3}) \right) \\ &= \sum_{i=0}^{3} x_{i,0} + x_{i,1} + x_{i,2} + \sum_{i=0}^{3} \left(x_{i,3} \sum_{j=0}^{3} x_{j,2} + x_{i,2} \sum_{\ell=0}^{3} x_{\ell,3} + x_{i,2} x_{i,3} \right),\end{aligned}$$

where we obtain the last equality by replacing, for all t, $\sum_{j=0, j\neq i}^{3} x_{j,t}$ by $\left(\sum_{j=0}^{3} x_{j,t}\right) + x_{i,t}$ and then by simplifying thanks to cancellation modulo 2. Finally, by decomposing the second sum, we observe that:

$$\begin{aligned}\Sigma_{\text{col}} &= \sum_{i=0}^{3} x_{i,0} + x_{i,1} + x_{i,2} + \sum_{j=0}^{3} x_{j,2} \sum_{i=0}^{3} x_{i,3} + \sum_{\ell=0}^{3} x_{\ell,3} \sum_{i=0}^{3} x_{i,2} + \sum_{i=0}^{3} x_{i,2} x_{i,3} \\ &= \sum_{i=0}^{3} x_{i,0} + x_{i,1} + x_{i,2} + \sum_{i=0}^{3} x_{i,2} x_{i,3} = \sum_{i=0}^{3} g(x_i).\end{aligned}$$

We therefore deduce that $(x_0, x_1, x_2, x_3) \mapsto \sum_{i=0}^{3} g(x_i)$ is an invariant for M, and therefore that $\widetilde{g}$ is an invariant of the whole MixColumns MC. We prove in a similar manner that $\widetilde{f}$ is an invariant of MC. In the original paper [TLS19, Theorem 1], this result is proved even more generally for quadratic invariants and orthogonal linear layers.

Only the addition of a constant (or key) remains to be studied. Let $c \in \mathbb{F}_2^4$. For any $x \in \mathbb{F}_2^4$, we get:

$$\begin{aligned} g(x+c) &= (x_0+c_0)+(x_1+c_1)+(x_2+c_2)+(x_2+c_2)(x_3+c_3) \\ &= g(x)+g(c)+x_2c_3+x_3c_2. \end{aligned}$$

Therefore, if $c_2 = c_3 = 0$, that is, if $c \in \langle \mathtt{0x1}, \mathtt{0x2} \rangle$, then $g(x+c) = g(x)+g(c)$. This is in particular the case of *each nibble of the round constants* of Midori64. This implies that $\widetilde{g}$ is an invariant of the constant additions satisfying:

$$\forall\ r \in [\![0, R-2]\!]\,, \forall\ x \in \mathbb{F}_2^{64}, \widetilde{g}(x+c^{(r)}) = \widetilde{g}(x)+\widetilde{g}(c^{(r)}).$$

Non-linear invariant of Midori64. All in all, we deduce that for any $x \in \mathbb{F}_2^{64}$ and $r \in [\![0, R-2]\!]$:

$$\begin{aligned} \widetilde{g}(T_{c^{(r)}} \circ \mathsf{MC} \circ \mathsf{SC} \circ \mathcal{S}(x)) &= \widetilde{g}(\mathsf{MC} \circ \mathsf{SC} \circ \mathcal{S}(x)) + \widetilde{g}(c^{(r)}) \\ &= \widetilde{g}(\mathsf{SC} \circ \mathcal{S}(x)) + \widetilde{g}(c^{(r)}) \\ &= \widetilde{g}(\mathcal{S}(x)) + \widetilde{g}(c^{(r)}) \\ &= \widetilde{g}(x) + \widetilde{g}(c^{(r)}). \end{aligned}$$

Finally, *by assuming that all round keys $k^{(r)}$ satisfy $k_i^r \in \langle \mathtt{0x1}, \mathtt{0x2} \rangle$* for any $i \in [\![0, 15]\!]$, and that the same holds for the whitening key, a similar reasoning leads to:

$$\forall x \in \mathbb{F}_2^{(64)}, \quad \widetilde{g}(E_k(x)) = \widetilde{g}(x) + \sum_{r=0}^{R-2} \widetilde{g}(c^{(r)}) + \sum_{r=0}^{R-2} \widetilde{g}(k^{(r)}) + \widetilde{g}(\mathrm{WK}) + \widetilde{g}(\mathrm{WK}),$$

which means that the sum $\widetilde{g}(E_k(x)) + \widetilde{g}(x)$ is independent of the input message x. Note that it also provides one quadratic equation in the key bits. Because of the light key-schedule of Midori64 that is described in Section 4.1.2.b, we immediately observe that if both halves $k_0, k_1 \in \mathbb{F}_2^{64}$ of the master key satisfy $k_0, k_1 \in \langle \mathtt{0x1}, \mathtt{0x2} \rangle^{16}$, then all the aforementioned conditions on the whitening and round keys are satisfied. The space of weak keys is therefore $\langle \mathtt{0x1}, \mathtt{0x2} \rangle^{32}$, and has cardinality 2^{64}.

Non-linear invariant of the Vert family. The same analysis remains true for the whole families Vert^c where $c \in \langle \mathtt{0x1}, \mathtt{0x2} \rangle$, as the function $\widetilde{g}$ is an invariant of *any* ShuffleCells operation and of *any* round constant addition in that case.

Regarding $\widetilde{f}$, we observe that if, a constant $c \in \mathbb{F}_2^4$ satisfies $c_3 = 0$ and $c_0 = c_2$, that is, if $c \in \langle \mathtt{0x2}, \mathtt{0x5} \rangle$, then f is an invariant of the constant addition over $\mathbb{F}_2^4$ and $\widetilde{f}$ of the constant addition over $\mathbb{F}_2^{64}$ if the condition is satisfied for each nibble. Therefore, $\widetilde{f}$ is *not* an invariant of Midori64, as some nibbles of round constants can take the value 1, that does not belong to $\langle \texttt{0x2}, \texttt{0x5}\rangle$. However, as already mentioned in [TLS19], $\widetilde{f}$ is an invariant of any member of Vert^c, with $c \in \langle \texttt{0x2}, \texttt{0x5}\rangle$, with a space of weak keys of cardinality 2^{64}.

4.2.3 Non-linear invariants from two other points of view

Non-linear invariants of block ciphers were encompassed into two fruitful frameworks that were also applied to analyze Midori64.

4.2.3.a The matrix point of view

Invariants as eigenvectors. In [Bey18], Beyne studies the non-linear invariants of Midori64 from the matrix point of view that is presented in Section 2.2.2.b. As outlined in the aforementioned section, the transition matrix $\mathbf{T}^F$ of a function $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ is, by construction, (the matrix of) the operator which maps for any $x \in \mathbb{F}_2^n$, the function $\mathbf{1}_x\colon \mathbb{F}_2^n \to \mathbb{F}_2$ to the function $\mathbf{1}_{F(x)}$. This means that for any *complex-valued* Boolean function $g\colon \mathbb{F}_2^n \to \mathbb{C}$, we have, $\mathbf{T}^F(g) = g \circ F$. In particular for a Boolean function $f\colon \mathbb{F}_2^n \to \mathbb{F}_2$, by defining g as $g = (-1)^f$, we obtain $\mathbf{T}^F(-1)^f = (-1)^f \circ F = (-1)^{f\circ F}$. This enables us to reformulate Eq. (4.2) as:

$$(-1)^f = (-1)^{f\circ E_k + \varepsilon_k} \iff (-1)^f = (-1)^{\varepsilon_k}(-1)^{f\circ E_k} = (-1)^{\varepsilon_k}\mathbf{T}^{E_k}(-1)^f .$$

In other words, a Boolean function $f\colon \mathbb{F}_2^n \to \mathbb{F}_2$ is an invariant of E_k if and only if $(-1)^f$ is an eigenvector of $\mathbf{T}^F$ with eigenvalue ±1. In [Bey18], Beyne rather studies the eigenvectors of the correlation matrix $\mathbf{C}^F$ with eigenvalue ±1, which are in one-to-one correspondence with the latter ones, and manages to encounter again the aforementioned invariant subspace and non-linear invariants. But he also exhibits a non-linear invariant over two rounds of Midori64 which is not an invariant under a single round. After all, an eigenvector of a product of matrix does not have to be an eigenvector of each individual matrix, or equivalently the two-round invariant $f \xrightarrow{F\circ G} f$ does not necessarily mean that $f \xrightarrow{G} f$ and $f \xrightarrow{F} f$.

New invariants for Vert. To do so and by using his framework, Beyne exhibits two functions $h_0, h_1\colon \mathbb{F}_2^4 \to \mathbb{F}_2$ defined by:

$$h_0\colon (x_0,x_1,x_2,x_3) \mapsto x_0 + x_2, \qquad h_1\colon (x_0,x_1,x_2,x_3) \mapsto x_0x_2 + x_0 + x_1 + x_3 + 1,$$

which satisfy $h_1 \xrightarrow{S} h_0$. This can be immediately deduced from the ANF of S that is given in Eq. (4.3) by looking at $h_0 \circ S = S_0 + S_2$.

$$\begin{aligned} &S\colon (\mathbb{F}_2)^4 \to (\mathbb{F}_2)^4 \\ &\begin{pmatrix} x_0\\ x_1\\ x_2\\ x_3\end{pmatrix} \mapsto \begin{pmatrix} x_0x_1x_2 + x_0x_1x_3 + x_0x_2 + x_0x_3 + x_1x_2x_3 + x_1\\ x_0x_2 + x_0x_3 + x_0 + x_2x_3 + x_2\\ x_0x_1x_2 + x_0x_1x_3 + x_0x_3 + x_0 + x_1x_2x_3 + x_3 + 1\\ x_0x_1x_3 + x_0x_1 + x_1x_2x_3 + x_1x_3 + x_2x_3 + 1\end{pmatrix}. \end{aligned} \tag{4.3}$$

By defining $\widetilde{h_0}, \widetilde{h_1} : \mathbb{F}_2^{64} \to \mathbb{F}_2$ in a similar manner to Section 4.2.2, and reasoning likewise, we prove that for any $c \in \mathbb{F}_2^{64}$, it holds that:

$$\widetilde{h_0} \circ \mathcal{S} = \widetilde{h_1}, \quad \widetilde{h_0} \circ \mathsf{MC} \circ \mathsf{SC} = \widetilde{h_0}, \text{ but also } \quad \widetilde{h_0} \circ T_c = \widetilde{h_0} + \widetilde{h_0}(c).$$

The fact that the last equality holds without any restriction on c is a consequence of the linearity of h_0. All in all, we obtain:

$$\forall c \in \mathbb{F}_2^{64}, \quad \widetilde{h_0} \circ T_c \circ \mathsf{MC} \circ \mathsf{SC} \circ \mathcal{S} = \widetilde{h_1} + \widetilde{h_0}(c).$$

Furthermore, because S is involutive, we immediately observe that $h_0 \circ S = h_1$ is equivalent to $h_1 \circ S = h_0$. We therefore prove in a similar manner that:

$$\forall c \in \langle \mathtt{0x2}, \mathtt{0x8} \rangle^{16}, \quad \widetilde{h_1} \circ T_c \circ \mathsf{MC} \circ \mathsf{SC} \circ \mathcal{S} = \widetilde{h_0} + \widetilde{h_1}(c),$$

where the condition on c is due to the non-linearity of h_1 and is deduced as in Section 4.2.2. This proves that the following two equalities hold for all $c_0 \in \mathbb{F}_2^{64}$ and all $c_1 \in \langle \mathtt{0x2}, \mathtt{0x8} \rangle^{16}$:

$$\widetilde{h_0} \circ T_{c_0} \circ \mathsf{MC} \circ \mathsf{SC} \circ \mathcal{S} \circ T_{c_1} \circ \mathsf{MC} \circ \mathsf{SC} \circ \mathcal{S} = \widetilde{h_0} + \widetilde{h_0}(c_0) + \widetilde{h_1}(c_1), \tag{4.4}$$

$$\widetilde{h_1} \circ T_{c_1} \circ \mathsf{MC} \circ \mathsf{SC} \circ \mathcal{S} \circ T_{c_0} \circ \mathsf{MC} \circ \mathsf{SC} \circ \mathcal{S} = \widetilde{h_1} + \widetilde{h_0}(c_0) + \widetilde{h_1}(c_1). \tag{4.5}$$

This gives two invariants of two rounds of Midori, with conditions only on a single key/constant addition, which provides distinguishers for the family Vert^c where $c \in \langle \mathtt{0x2}, \mathtt{0x8} \rangle$, but not for the original Midori64 as $1 \notin \langle \mathtt{0x2}, \mathtt{0x8} \rangle$. The weak-key space is of cardinality 2^{96} as 32 linear conditions have to be satisfied, but only by one of the two halves of the master key k.

Weak-key exact linear approximations. Very interestingly, Eq. (4.4) can be interpreted from the point of view of linear cryptanalysis. Indeed, because $\widetilde{h_0}$ is linear, Eq. (4.4) is nothing else than an *exact* linear approximation, *i.e.* that holds with probability 1, in a weak-key setting. This phenomenon goes against intuition for at least two reasons.

- First, such weak-key exact linear approximations exist for *any* even number of rounds. This contradicts, at least in a weak-key setting, the preconception that the higher the number of rounds is, the more secure the cipher is.

- Secondly, this holds even if the correlation of each linear trail (see Eq. (2.18)) is proved in the original paper [Ban+15] to be of low absolute value. As mentioned in the comments following Proposition 2.46, this is reminiscent of a clustering effect where *all* trails have the *same sign* and therefore leads to a maximal correlation.

These two counter-intuitive points are again put on the table with our analyzes in Section 4.3.3 but also in Section 5.5.2, but this time with a differential flavor. Similarly, we also show in the aforementioned section that commutative cryptanalysis provides a tool for capturing differential clustering effects, in the same way as non-linear invariants capture linear clustering.

4.2.3.b Conjugate-cipher point of view

In a subsequent work, Beierle, Canteaut & Leander [BCL18] continued investigating the link between linear cryptanalysis and invariants.

Invariants and high absolute Walsh coefficients. First, they prove in [BCL18, Theorems 4 and 5] that the existence of quadratic invariants, or of invariants subspaces, as in the case of Midori or Vert, always imply the existence of highly-biased linear approximations for the whole cipher instantiated with a weak key. In particular, the lower bound on the absolute value of the corresponding Walsh coefficient $W_{E_k}(\alpha, \beta)$ is exact and not based on any heuristic such as dominant trails. It can by itself contradicts claims made by some designers. The black spot is however that this result is non-constructive: an attacker would still have to find such a linear approximation, which can *a priori* be different for all keys.

Conjugate ciphers with linear flaws. Furthermore, the authors also unveil a way to study *balanced* non-linear invariants through the scope of linear cryptanalysis. To do so, it suffices to encapsulate the balanced Boolean function $g\colon \mathbb{F}_2^n \to \mathbb{F}_2$ as the α-component of a bijection $G\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$: $\alpha \cdot G = g$. Such a G can be built as presented in Example 4.3 below. Then, using the bijectivity of G, we can observe that:

$$\begin{aligned}\forall x \in \mathbb{F}_2^n, \quad g(x) + g(E_k(x)) = \varepsilon_k \iff{} & \forall x \in \mathbb{F}_2^n,\ \alpha \cdot G(x) + \alpha \cdot G(E_k(x)) = \varepsilon_k \\ \iff{} & \forall x \in \mathbb{F}_2^n,\ \alpha \cdot x + \alpha \cdot G(E_k(G^{-1}(x))) = \varepsilon_k .\end{aligned}$$

In other words, the *conjugate cipher* $\mathcal{E}^G := (G \circ E_k \circ G^{-1})_{k \in \mathbb{F}_2^\kappa}$ admits a space of weak keys for which there exists a linear relation in input and output bits that holds with probability 1. Equivalently, the existence of a non-linear invariant can be understood as the existence of an invariant affine hyperplane of some conjugate ciphers. In the following, we denote by F^G the conjugate of a function $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ by a bijection $G\colon \mathbb{F}_2^n \xrightarrow{\sim} \mathbb{F}_2^n$, that is, $F^G := G \circ F \circ G^{-1}$.

Example 4.3 (Construction of G)**.** Let $g\colon \mathbb{F}_2^n \to \mathbb{F}_2$ be a balanced Boolean function and let us build $G\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ such that $G_0 = g$. This particular case corresponds to $\alpha = \texttt{0x1}$ and is actually sufficient to consider. Indeed, for any linear bijection L, the sets of components of G and of $L \circ G$ are identical, but each individual component is located at a different place as $\alpha \cdot L \circ G = L^\top(\alpha) \cdot G$. For G with $G_0 = g$ to be balanced, it is necessary and sufficient that the restriction of G to its $n-1$ last variables, $\pi_{n-1} \circ G\colon \mathbb{F}_2^n \to \mathbb{F}_2^{n-1}$ is balanced on both $\mathrm{Supp}(g)$ and $\mathbb{F}_2^n \setminus \mathrm{Supp}(g)$. In order to build G, we can look at its LUT as a matrix $M \in \mathbf{M}_{n \times 2^n}(\mathbb{F}_2)$ where each row is the look-up table of one of its coordinates, the first one being already set. Then, M can be completed by considering the two disjoint submatrices $P := M_{[\![1,n-1]\!] \times \mathrm{Supp}(g)}$ and $Q := M_{[\![1,n-1]\!] \times (\mathbb{F}_2^n \setminus \mathrm{Supp}(g))}$. Because P and Q have the size of the LUT of vectorial Boolean functions with $n-1$ inputs and outputs bits, replacing P and Q by the LUT of any bijection of $\mathbb{F}_2^{n-1}$ leads to a bijection G of $\mathbb{F}_2^n$ with the announced property. ▷

This point of view provides interesting directions. First, we observe that invariants only study situations where the input and output masks are both equal (to α). In standard linear cryptanalysis, such a restriction does not hold, and should not hold here as well. By considering possibly different masks α, β, an exact linear approximation for $\mathcal{E}^G$ coincides with the existence of two balanced Boolean functions $f, g\colon \mathbb{F}_2^n \to \mathbb{F}_2$ such that $f = g \circ E_k$ for many k. This is already mentioned by Todo, Leander & Sasaki [TLS19] as a direction "clearly worth discussing", while the 2-round invariant of Beyne [Bey18] that is presented in Section 4.2.3.a is based on such a property over one round. Moreover, this framework enables us to study probabilistic linear approximations of $\mathcal{E}^G$ and therefore probabilistic non-linear invariants of $\mathcal{E}$ using the classical methods of linear cryptanalysis. For instance, by decomposing any E_k^G as:

$$E_k^G = G \circ E_k \circ G^{-1} = \underbrace{G \circ F_k^{(R-1)} \circ G^{-1}}_{\left(F_k^{(R-1)}\right)^G} \circ G \circ F_k^{(R-2)} \circ \cdots \circ \underbrace{G \circ F_k^{(0)} \circ G^{-1}}_{\left(F_k^{(0)}\right)^G}, \tag{4.6}$$

the conjugates of the round functions can be studied in order to find some dominant linear trails.

Linear cryptanalysis of a conjugate of Midori. Such a study is already presented in [BCL18]. In particular, the authors point out some surprising phenomenon.

First of all, the aforementioned invariants are all of the form $\widetilde{f}\colon (\mathbb{F}_2^4)^{16} \to \mathbb{F}_2, x \mapsto \sum_{i=0}^{15} f(x_i)$, for some invariant $f\colon \mathbb{F}_2^4 \to \mathbb{F}_2$ of the Sbox S. Because it is built iteratively, this implies that the corresponding linear approximation $\alpha \cdot x = \alpha \cdot E_k^G(x) + \varepsilon_k$, not only has probability 1, it also corresponds to a very degenerate case of Proposition 2.46 where the linear trail $\alpha \to \alpha \to \dots \alpha$ is the only trail with non-zero correlation. Furthermore, as each intermediate mask is equal to $\alpha = (a, a, \dots, a) \in (\mathbb{F}_2^4)^{16}$, for some $a \in \mathbb{F}_2^4 \setminus \{0\}$, this also means, using the standard vocabulary, that *each Sbox of each round is linearly active.* This does not contradict the wide-trail strategy arguments, as this holds for a *non-linear* conjugate $\mathcal{E}^G$, and not for $\mathcal{E}$. Indeed, contrary to linear equivalence (see Proposition 2.64), non-linear conjugation does not preserve linearity: while $\mathcal{L}(S) = 8$, S^G is by design built so that $\mathcal{L}(S^G) = 16$. This is nevertheless remarkable, as a related observation in the differential setting is made in Section 4.3.3.

Furthermore, the probabilistic case is hard to handle. As an example, it is clear by definition that the number of solutions of $\alpha \cdot x \approx \beta \cdot E_k^G(x)$ or equivalently of $\alpha \cdot G(x) \approx \beta \cdot G(E_k(x))$ only depends on the two components $\alpha \cdot G$ and $\beta \cdot G$. However, by using Proposition 2.46 to approximate this number of solutions, we observe that the correlation of a trail may depend on all the other components. As shown by the authors, the approximation of the number of solutions using multiple trails can, and in practice does, depend on the choice of change of variables G. This implies that some choices of G theoretically provide an easier understanding of the possible clustering effect, but no heuristic for this choice is presently known.

x	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
$h(x)$	0	1	1	0	1	0	0	1	1	0	1	0	0	1	1	0
$S(x)$	c	a	d	3	e	b	f	7	8	9	1	5	0	2	4	6
$h(S(x))$	0	1	1	0	1	0	0	1	1	0	1	0	0	1	1	0

Table 4.5: The cubic invariant h of the Sbox of Midori64.

Nevertheless, the correlation of any single trail with intermediate masks $\gamma = (\gamma_i)_{i\in[\![0,15]\!]}$, where $\gamma_i \in \{0, a, b\}$ with $\alpha =: (a, \dots, a)$ and $\beta =: (b, \dots, b)$, is only dependent on the two coordinates $\alpha \cdot G$ and $\beta \cdot G$. An approximation of $W_{E_k^G}(\alpha, \beta)$ can thus be given without a formal instantiation of G. Such a study is made in [BCL18, Section 4.2] by considering deterministic transitions through the Sbox layer and probabilistic transitions through the linear layer. This provides a quite accurate estimation of the experimentally observed probability. The obtained distinguisher over four rounds is probabilistic, but has the advantage to hold for a bigger space of weak keys. In [BCL18, Section 4.3], some of the deterministic Sbox transitions are replaced by probabilistic ones to overall obtain a bigger estimated probability over one full round, and further, over the full Midori. However the authors experimentally show that the observed probability is actually highly key-dependent and that the estimated average over all keys is not as representative as one might at first expect. This is highlighted with a 2-round toy example over a 16-bit state that is described as:

$$M \circ T_{k_1} \circ (S^{\times 4}) \circ M \circ T_{k_0} \circ (S^{\times 4}),$$

where $k_0, k_1 \in \mathbb{F}_2^{16}$ are independent round keys, $M\colon (\mathbb{F}_2^4)^4 \to (\mathbb{F}_2^4)^4$ is the MixColumns matrix of Midori and $S^{\times 4}\colon (\mathbb{F}_2^4)^4 \to (\mathbb{F}_2^4)^4$ is defined by $(x_0, x_1, x_2, x_3) \mapsto (S(x_0), S(x_1), S(x_2), S(x_3))$. The invariant that is considered is based on the already-introduced invariant for S, $g\colon \mathbb{F}_2^4 \to \mathbb{F}_2, x \mapsto x_0 + x_1 + x_2 + x_2x_3$ and a new one $h\colon \mathbb{F}_2^4 \to \mathbb{F}_2$ that is defined by:

$$h\colon x \mapsto x_1x_2x_3 + x_1x_3 + x_0 + x_1 + x_2 + x_3.$$

The look-up table of h is given in Table 4.5. Then, the invariant of the toy cipher is denoted by $\rho\colon (\mathbb{F}_2^4)^4 \to \mathbb{F}_2$ and defined by:

$$\rho\colon (x_0, x_1, x_2, x_3) \mapsto h(x_0) + g(x_1) + g(x_2) + g(x_3).$$

The linear trail corresponding to the natural transitions $\rho \xrightarrow{T_k \circ S^{\times 4}} \rho \xrightarrow{M} \rho$ was believed to be dominant and its absolute correlation of at least $9/32$ to be a good approximation of the correlation of the approximation $\rho \circ M \circ T_k \circ S^{\times 4} \approx \rho$. This is the reason why the absolute correlation of $\rho \circ M \circ T_{k_1} \circ S^{\times 4} \circ M \circ T_{k_0} \circ S^{\times 4} \approx \rho$ was believed to be at least equal to $(9/32)^2$. Instead, using his matrix-based framework, Beyne shows in [Bey21, Section 7] that under some conditions on k_1, there exists a better trail with correlation at least $9/16$. It is given by:

$$\rho \xrightarrow{T_{k_0} \circ S^{\times 4}} \tau \xrightarrow{M} \tau \xrightarrow{T_{k_1} \circ S^{\times 4}} \tau \xrightarrow{M} \rho,$$

where $\tau\colon (\mathbb{F}_2^4)^4 \to \mathbb{F}_2$ is defined by $\tau\colon x_0, x_1, x_2, x_3 \mapsto \sum_{i=0}^{3} g(x_i)$. With the benefit of hindsight, this trail is not so surprising. Indeed, as shown in Tables 4.4 and 4.5, g and h only differ for two inputs. This implies that $\tau(S^{\times 4}(x)) = \rho(x)$ holds with probability 14/16. Swapping from the probability-1 transition $\rho \xrightarrow{S^{\times 4}} \rho$ to the probabilistic transition $\rho \xrightarrow{S^{\times 4}} \tau$, has the advantage to now consider τ, which is an invariant of the first linear layer and the second Sbox layer. The mentioned conditions on k_1 are the ones that allow $\tau \xrightarrow{T_{k_1}} \tau$ to hold with probability 1. They correspond to the conditions given for g in Section 4.2.2. This explains part of the observations of [BCL18]. More of them are addressed in [Bey21, Section 7.2].

4.3 Differential cryptanalysis of conjugate ciphers

If previous works have thoroughly studied linear cryptanalysis of conjugate ciphers, the differential one has been (to the best of our knowledge) left out until now. As we show in this section, some specific conjugate ciphers can also have some differential flaws. The Vert family, and in particular Midori64, serves as an example throughout the section. This work, which is interesting by itself, is also the starting point of the commutative framework that is presented in the next chapter.

4.3.1 Selecting interesting conjugates

4.3.1.a General overview

Let $\mathcal{E} = (E_k\colon \mathbb{F}_2^n \xrightarrow{\sim} \mathbb{F}_2^n)$ be a key-alternating block cipher where E_k is iteratively built as the composition of the round functions $(F_k^{(r)})_{r\in[\![0,R-1]\!]}$, *i.e.*:

$$E_k = F_k^{(R-1)} \circ \ldots \circ F_k^{(0)}.$$

As highlighted by Eq. (4.6), any conjugate cipher E_k^G can be considered as the composition of conjugate round functions $(F_k^{(r)})^G$. This enables us to perform a round-by-round analysis of its differential properties. From the functional point of view, this amounts to study the block cipher, up to a change of variables described by G. However, in order to be relevant and practical, some specific choices of G need to be made.

- First, G needs to be a *non-linear* bijection. Indeed, as mentioned in Proposition 2.64, the differential properties of two affine-equivalent functions are identical, and little can therefore be expected from the consideration of affine changes of variables.

- Secondly, as nothing is *a priori* known about the differential properties of non-linear conjugates, the class of changes of variables that are considered must be *manageable*. We therefore restrict ourselves to the parallel application of changes of variables at the cell level. Stated otherwise, we consider the cases of $G\colon \mathbb{F}_2^m \xrightarrow{\sim} \mathbb{F}_2^m$ where m is the size of the Sbox, and look at conjugates ciphers of the form $\mathcal{E}^{\mathcal{G}}$ where $\mathcal{G}\colon (\mathbb{F}_2^m)^s \xrightarrow{\sim} (\mathbb{F}_2^m)^s$ is the s-time parallel application of G, with $n = m \times s$. This in particular enables us to study the Sbox layer by only focusing on the conjugate S^G of a single Sbox.

- Finally, as shown in Proposition 2.33, the differential study of linear layers and constant additions are usually easy to handle. However, when conjugated by a non-linear bijection, those layers become *a priori* non-linear and their study is therefore more intricate. In particular, the reasoning presented in Section 2.3.3.d, and the standard assumptions of key-independence of Section 2.3.3.e do not hold because a conjugate constant addition T_c^G, with $c \in \mathbb{F}_2^m$, is considered as a whole, and not as the successive composition of G^{-1}, T_c and G. We therefore study T_c^G in a weak-key setting. As we mention below, keeping G as simple as possible simplifies this study.

These remarks have to be taken with caution. Regarding the restriction to *non-linear* change of variables, we note that even linear ones can sometimes be meaningful. For instance, in the already-mentioned work of Leurent & Pernot [LP21], a linear change of variables for the key schedule of the AES is exhibited. It also highly simplifies the overall structure of this component and leads to a better understanding of how it operates.

Regarding the third point, a counter-example that can be given is the case of a permutation of cells $\mathcal{P}\colon (\mathbb{F}_2^m)^s \xrightarrow{\sim} (\mathbb{F}_2^m)^s$. Indeed, applying in parallel the same mapping G before or after a permutation of cells is equivalent: $\mathcal{G} \circ \mathcal{P} = \mathcal{P} \circ \mathcal{G}$. In other words, it holds that $\mathcal{P}^{\mathcal{G}} = \mathcal{P}$, which means that the conjugate permutation remains identical, and in particular linear, no matter the complexity of G.

4.3.1.b The actual explored space for Midori64

Given the analysis sketched above, we start investigating the case of Midori by looking at conjugates of its S-box S. With the necessity for G to be simple and sparse enough, we consider $G\colon \mathbb{F}_2^4 \to \mathbb{F}_2^4$ such that it only contains linear coordinates except for a single one which is non-linear. These choices are not only the “simplest” choices of non-linear mappings G, they are also in line with some of the choices made by Beierle, Canteaut & Leander [BCL18]. Indeed, in [BCL18, Section 4.3] the mapping that is denoted by G_1' has a coordinate which coincides with the previously-introduced invariant h, while the three other coordinates are linear.

Following this direction, we choose at first to study bijections of the form:

$$G_g(x_0, x_1, x_2, x_3) := \big(x_0 + g(x_1, x_2, x_3), x_1, x_2, x_3\big) \ ,$$

where g is a (non-linear) Boolean function in variables x_1, x_2, x_3. Such bijections are in fact involutive and correspond to Feistel-like structures. The components of such functions can be partitioned into a subspace $V = \langle \texttt{0x2}, \texttt{0x4}, \texttt{0x8} \rangle$ corresponding to 2^3 affine components and an affine space $\texttt{0x1} + V$ of non-linear components. In that sense, this choice of G_g induces an alignment with the canonical basis. Our search space is nonetheless not limited by this property. Indeed, such an alignment disappears as soon as we compose G_g with a linear bijection L_{out} in output. However, such a composition sums up to study the conjugate $L_{\text{out}} \circ G_g \circ S \circ G_g \circ L_{\text{out}}^{-1}$, which has the same differential property as the linear-equivalent mapping $G_g \circ S \circ G_g$.

Another arbitrary choice that appears is the fact that G_g is necessarily linear in x_0. This can be solved by composing G_g with a linear bijection L_{in} in input. In that case, the mapping $S^{G_g \circ L_{\text{in}}} := G_g \circ L_{\text{in}} \circ S \circ L_{\text{in}}^{-1} \circ G_g$ is studied, or stated otherwise, the linear-equivalent Sbox $S^{L_{\text{in}}}$ is studied up to the non-linear change of variables G_g. Note that while $S^{L_{\text{in}}}$ and S share the same differential properties, this is *a priori* not the case for S^{G_g} and $S^{G_g \circ L_{\text{in}}}$.

The choice of L_{in}^{-1} was limited to the usage of the following simple deterministic algorithm. First, the image of `0x1`, *i.e.* $\xi^{(0)}$, is freely chosen, and then the images of `0x2`, `0x4`, `0x8`, *i.e.* $\xi^{(1)}, \xi^{(2)}, \xi^{(3)}$, are successively chosen as the minimum value such that the rank of the partial list of images increases. The image of the canonical basis is therefore a basis of $\mathbb{F}_2^4$, from which a linear mapping L_{in}^{-1} is obtained by linearly expanding the definition. We denote such a mapping by L_a^{-1}, where a is the image of `0x1`.

All in all, we focus on the changes of variables $G_{g,a}$ that are defined by: $G_{g,a} := G_g \circ L_a^{-1}$. This space is sufficiently constrained to be efficiently explored in practice. Indeed, $2^4 - 1$ choices can be made for a, while there exist 2^8 Boolean functions g mapping three bits to one. This gives about 2^{12} possibilities for $G_{g,a}$.

4.3.2 Layer by layer analysis

The mentioned space of conjugates was first filtered by analyzing the conjugates of the Sbox. A layer-by-layer analysis was then conducted with the promising conjugates of Midori. This process is described in detail in the following subsections.

4.3.2.a Sbox layer

Within the class presented above, many changes of variables G induce very weak conjugates of the S-box S of Midori. More precisely, we identified bijections G such that S^G has a probability-one differential $\Delta \to \Delta$.

For example, the look-up tables of $G_{a,g}$ and $S^{G_{a,g}}$, in the case where $g(x_1, x_2, x_3) = x_1 + x_1 x_3 = x_1(x_3 + 1)$, $a =$ `0x5`, and $\Delta =$ `0xd`, are given in Table 4.6, and the matrices of L_a and L_a^{-1} given in Eq. (4.7).

x	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
$G_{a,g}(x)$	0	3	4	7	2	1	6	5	8	a	c	e	b	9	f	d
$S^{G_{a,g}}(x)$	b	e	f	c	9	5	d	7	8	4	a	0	3	6	1	2
$S^{G_{a,g}}(x+\Delta)$	6	3	2	1	4	8	0	a	5	9	7	d	e	b	c	f

Table 4.6: A specific change of variables for the Sbox of Midori64.

$$L_a = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad L_a^{-1} = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{4.7}$$

As we can easily verify in that case, it holds that $D_\Delta S^{G_{a,g}}(x) = \Delta$ for any $x \in \mathbb{F}_2^4$. This immediately yields a probability-one $\nabla \to \nabla$ transition through the conjugate of the S-box layer $\mathcal{S}^{\mathcal{G}}$, where $\nabla := (\Delta, \dots, \Delta) \in (\mathbb{F}_2^4)^{16}$ and $\mathcal{G} = G_{a,g}^{\times 16}$.

4.3.2.b Constant addition

Let us now focus on the other layers. First, we observe that for any $x, c \in \mathbb{F}_2^4$, and any $g \colon \mathbb{F}_2^3 \to \mathbb{F}_2$ it holds that:

$$\begin{aligned} T_c^{G_g}(x) &= G_g \begin{pmatrix} x_0 + c_0 + g(x_1, x_2, x_3) \\ x_1 + c_1 \\ x_2 + c_2 \\ x_3 + c_3 \end{pmatrix} \\ &= \begin{pmatrix} x_0 + c_0 + g(x_1, x_2, x_3) + g(x_1 + c_1, x_2 + c_2, x_3 + c_3) \\ x_1 + c_1 \\ x_2 + c_2 \\ x_3 + c_3 \end{pmatrix} \\ &= T_c \circ G_{D_\delta g}(x), \end{aligned} \tag{4.8}$$

where $\delta = (c_1, c_2, c_3)$.

In the same way, because $L_a \circ T_c \circ L_a^{-1}(x) = L_a(L_a^{-1}(x) + c) = x + L_a(c) = T_{L_a(c)}(x)$, we observe that:

$$T_c^{G_{a,g}} = G_g \circ T_{L_a(c)} \circ G_g = T_{L_a(c)} \circ G_{D_\delta g}, \tag{4.9}$$

where $\delta \in \mathbb{F}_2^3$ corresponds in that case to the last three coordinates of $L_a(c)$.

In particular, if g is quadratic (as it is the case for the specific one given in Table 4.6), its derivative $D_\delta g$ is of degree at most 1, and so does $G_{D_\delta g}$ and therefore $T_c^{G_{a,g}} = T_{L_a^{-1}(c)} \circ G_{D_\delta g}$. This implies in that case that for any constant $c \in \mathbb{F}_2^4$, the derivative of $T_c^{G_{a,g}}$ is constant, which means that any differential $\Delta^{\text{in}} \xrightarrow{T_c^{G_{a,g}}} \Delta^{\text{out}}$ is deterministic, and has probability 0 or 1.

In our case, we denote by $W(\Delta, \Delta)$ the set of constants for which $\Delta \xrightarrow{T_c^{G_{a,g}}} \Delta$ holds with probability 1, that is:

$$W(\Delta, \Delta) := \left\{ c \in \mathbb{F}_2^4,\ D_\Delta T_c^{G_{a,g}}(x) = \Delta \ \forall\ x \in \mathbb{F}_2^4 \right\}. \tag{4.10}$$

In the case where g is quadratic, the definition of $W(\Delta, \Delta)$ can be simplified into:

$$W(\Delta, \Delta) = \left\{ c \in \mathbb{F}_2^4,\ D_\Delta T_c^{G_{a,g}}(0) = \Delta \right\}.$$

We now determine $W(\Delta, \Delta)$ in the specific case where g is chosen as in Table 4.6, and where $\Delta = \texttt{0xd} = \texttt{0b1101}$, $a = \texttt{0x5}$.

By replacing g and a in Eq. (4.9), we observe that for any $x \in \mathbb{F}_2^4$, it holds that:

$$\begin{aligned} T_c^{G_{a,g}}(x) &= \begin{pmatrix} x_0 + c_2 + x_1(x_3+1) + (x_1 + c_0 + c_2)(x_3 + c_3 + 1) \\ x_1 + c_0 + c_2 \\ x_2 + c_1 \\ x_3 + c_3 \end{pmatrix} \\ &= \begin{pmatrix} x_0 + x_1 c_3 + x_3(c_0 + c_2) + c_2 + (c_0 + c_2)(c_3 + 1) \\ x_1 + c_0 + c_2 \\ x_2 + c_1 \\ x_3 + c_3 \end{pmatrix}, \end{aligned} \tag{4.11}$$

where the ANF of $L_a(c)$ is immediately deduced from its matrix in Eq. (4.7). In particular, we can express $T_c^{G_{a,g}}(0)$ and $T_c^{G_{a,g}}(\Delta)$ as:

$$\begin{aligned} T_c^{G_{a,g}}(0) &= \begin{pmatrix} c_2 + (c_0 + c_2)(c_3 + 1) \\ c_0 + c_2 \\ c_1 \\ c_3 \end{pmatrix}, \\ T_c^{G_{a,g}}(\Delta) &= \begin{pmatrix} 1 + 0 \cdot c_3 + 1 \cdot (c_0 + c_2) + c_2 + (c_0 + c_2)(c_3 + 1) \\ 0 + c_0 + c_2 \\ 1 + c_1 \\ 1 + c_3 \end{pmatrix} \\ &= \begin{pmatrix} 1 + c_2 + (c_0 + c_2)c_3 \\ c_0 + c_2 \\ 1 + c_1 \\ 1 + c_3 \end{pmatrix}. \end{aligned}$$

Therefore, we can express $D_\Delta T_c^{G_{a,g}}(0)$ as:

$$D_\Delta T_c^{G_{a,g}}(0) = T_c^{G_{a,g}}(0) + T_c^{G_{a,g}}(\Delta) = \begin{pmatrix} 1 + c_0 + c_2 \\ 0 \\ 1 \\ 1 \end{pmatrix}, \tag{4.12}$$

which implies that $D_\Delta T_c^{G_{a,g}}(0) = \Delta$ if and only if $c_0 + c_2 + 1 = 1$, *i.e.* $c_0 = c_2$. In other words, we get $W(\Delta, \Delta) = \langle \texttt{0x2}, \texttt{0x5}, \texttt{0x8} \rangle$. To conclude, provided that a 64-bit round key or round constant C satisfies $C \in (W(\Delta, \Delta))^{16}$, there exists a probability-one differential $\nabla \to \nabla$ through the conjugate of key/constant addition $T_C^{\mathcal{G}}$.

4.3.2.c Linear layer

Let us now consider the linear layer and start with the MixColumns matrix M of Midori. Given any linear bijection $L\colon \mathbb{F}_2^4 \to \mathbb{F}_2^4$, and $(x_0, x_1, x_2, x_3) \in (\mathbb{F}_2^4)^4$ it holds that:

$$\begin{aligned}
M^{L^{\times 4}}(x_0, x_1, x_2, x_3) &= L^{\times 4} \circ M \circ (L^{-1})^{\times 4}(x_0, x_1, x_2, x_3) \\
&= L^{\times 4} \circ M(L^{-1}(x_0), L^{-1}(x_1), L^{-1}(x_2), L^{-1}(x_3)) \\
&= L^{\times 4}(L^{-1}(x_1 + x_2 + x_3), \ldots, L^{-1}(x_0 + x_1 + x_2)) \\
&= L^{\times 4} \circ (L^{-1})^{\times 4}(x_1 + x_2 + x_3, \ldots, x_0 + x_1 + x_2) \\
&= (x_1 + x_2 + x_3, \ldots, x_0 + x_1 + x_2) \\
&= M(x_0, x_1, x_2, x_3),
\end{aligned}$$

where the third equality comes from the definition of M and the linearity of L^{-1}.

In particular, the mapping $M^{G_{a,g}^{\times 4}}$ can be simplified as:

$$M^{G_{a,g}^{\times 4}} = G_g^{\times 4} \circ L_a^{\times 4} \circ M \circ (L_a^{-1})^{\times 4} \circ G_g^{\times 4} = G_g^{\times 4} \circ M \circ G_g^{\times 4} = M^{G_g^{\times 4}}.$$

Let us consider the first 4-bit coordinate of $M^{G_g^{\times 4}}$. We denote it by $N_0\colon (\mathbb{F}_2^4)^4 \to \mathbb{F}_2^4$, in other words, we have:

$$\begin{aligned}
N_0(x, y, z, t) &:= G_g(G_g(y) + G_g(z) + G_g(t)) \\
&= \begin{pmatrix} y_0 + z_0 + t_0 + g(y) + g(z) + g(t) + g(y + z + t) \\ y_1 + z_1 + t_1 \\ y_2 + z_2 + t_2 \\ y_3 + z_3 + t_3 \end{pmatrix}.
\end{aligned}$$

Therefore, its derivative $D_{(\Delta,\ldots,\Delta)}N_0$ can be expressed as:

$$\begin{aligned}
D_{(\Delta,\ldots,\Delta)}N_0(x, y, z, t) &= \begin{pmatrix} 3\Delta_0 + D_\delta g(y) + D_\delta g(z) + D_\delta g(t) + D_\delta g(y + z + t) \\ 3\Delta_1 \\ 3\Delta_2 \\ 3\Delta_3 \end{pmatrix} \\
&= \begin{pmatrix} \Delta_0 + D_\delta g(y) + D_\delta g(z) + D_\delta g(t) + D_\delta g(y + z + t) \\ \Delta_1 \\ \Delta_2 \\ \Delta_3 \end{pmatrix},
\end{aligned}$$

where δ is made of the 3 last coordinates of Δ. For the specific g given in Table 4.6, we observe that:

$$D_\delta g(x_1, x_2, x_3) = \delta_1(x_3 + 1) + \delta_3 x_1 + \delta_1\delta_3.$$

Furthermore, by considering Δ = `0xd` = `0b1101`, the expression becomes $D_\delta g(x_1, x_2, x_3) = x_1$. This implies in that case that:

$$D_\delta g(y) + D_\delta g(z) + D_\delta g(t) + D_\delta g(y + z + t) = y_1 + z_1 + t_1 + (y_1 + z_1 + t_1) = 0,$$

which means that $D_{(\Delta,\dots,\Delta)}N_0$ is constant and equal to Δ. The same reasoning on each of the three coordinates N_i with $i \in \{1,2,3\}$ that can be defined in the same way leads to the fact that $D_{(\Delta,\dots,\Delta)}M^{G_{a,g}^{\times 4}}$ is constant and equal to $(\Delta,\Delta,\Delta,\Delta)$, which ultimately means that $D_{\nabla}\mathsf{MC}^{\mathcal{G}} = \nabla$. Stated otherwise, the differential $\nabla \xrightarrow{\mathsf{MC}^{\mathcal{G}}} \nabla$ holds with probability 1.

4.3.2.d Permutation of cells

As already mentioned, for any permutation of cells $\mathcal{P}$, we have $\mathcal{P}^{\mathcal{G}} = \mathcal{P}$. This implies that the conjugate is linear and that $D_{\nabla}(\mathcal{P}^{\mathcal{G}})$ is a constant function equal to $\mathcal{P}(\nabla) = \nabla$. In other words, $\nabla \xrightarrow{\mathcal{P}^{\mathcal{G}}} \nabla$ holds with probability 1.

4.3.3 A new distinguisher for the **Vert** family

4.3.3.a A differential trail with probability 1

To sum up, provided that the round key and round constant belong to $(W(\Delta,\Delta))^{16}$, the differential $\nabla \to \nabla$ holds with probability 1 through all the layers of the *conjugate* round function of Midori. This implies that, *for any number of rounds* R, the iterated differential trail $\nabla \xrightarrow{(F_k^{(R-1)})^{\mathcal{G}}} \nabla \to \cdots \xrightarrow{F_k^{(0)})^{\mathcal{G}}} \nabla$ holds with probability 1, as long as all keys and constants belong to $(W(\Delta,\Delta))^{16}$. This *single* trail gives rise to a differential with probability one for the conjugate cipher $\mathcal{E}^{\mathcal{G}}$. This can be used as a distinguisher against the original cipher $\mathcal{E}$. Indeed, instead of querying chosen pairs of plaintexts the form $(x, x+\nabla)$, an adversary asks for the encryption of a pair $(\mathcal{G}^{-1}(x), (\mathcal{G}^{-1}(x+\nabla))$. With the corresponding pair of ciphertexts (y,z), he/she computes $\mathcal{G}(y)+\mathcal{G}(z)$ which should always be equal to ∇ in the case of $\mathcal{E}$. This happens with probability 2^{-n} in the case of a random function.

Because of the conditions on the round constants, this distinguisher does not work for the original Midori64. However, it works for any member of Vert^c for $c \in W(\Delta,\Delta) = \langle \mathtt{0x2}, \mathtt{0x5}, \mathtt{0x8} \rangle$, and for 2^{96} weak keys out of 2^{128}, as each nibble of the master key must belong to $W(\Delta,\Delta)$. This space of weak-key is, to the best of our knowledge, new.

4.3.3.b Relationship with previous distinguishers

Because of the close link between linear and differential cryptanalysis highlighted in Section 2.3.5, it is natural to question the relationship between this new distinguisher and the previous non-linear invariants that can be interpreted as linear approximations of $\mathcal{E}^{\mathcal{G}}$. The first thing to notice is that its space of weak keys strictly contains the weak keys of the invariant $x \mapsto x_0 + x_3 + x_0x_3 + x_2x_3$ exhibited by Todo, Leander & Sasaki [TLS19]. Indeed, this invariant exists if, as explained in Section 4.2.2, each nibble of the master key must lie in $\langle \mathtt{0x2}, \mathtt{0x5} \rangle \subset \langle \mathtt{0x2}, \mathtt{0x5}, \mathtt{0x8} \rangle$. On the other hand, our weak-key space has about the same cardinality as the one of the two-round invariant of Beyne [Bey18], see Section 4.2.3.a. Their intersection is non-trivial, but both sets are of different nature, as the one of Beyne is $\left(\langle \texttt{0x2}, \texttt{0x8}\rangle^{16} \times \mathbb{F}_2^{64}\right) \cup \left(\mathbb{F}_2^{64} \times \langle \texttt{0x2}, \texttt{0x8}\rangle^{16}\right)$ and it constrains only one half of the master key.

Some of these similarities can be explained by the specific $G_{a,g}$ that we use in the above section. The ANF of $G_{a,g}$ is given by:

$$G_{a,g}(x_0, x_1, x_2, x_3) \mapsto \begin{pmatrix} x_0x_3 + x_0 + x_2x_3 \\ x_0 + x_2 \\ x_1 \\ x_3 \end{pmatrix}.$$

As we can observe, the component $\texttt{0x9} \cdot G_{a,g}$ corresponds precisely to the non-linear invariant mentioned just above. Furthermore, it can be verified with Eq. (4.3) that $\texttt{0xf} \cdot G_{a,g} = \texttt{0x7} \cdot S + 1$. Let us define the linear function h_2 and the quadratic one h_3 by:

$$\begin{aligned} h_2(x) &:= \texttt{0x7} \cdot x = x_0 + x_1 + x_2, \\ h_3(x) &:= \texttt{0xf} \cdot G_{a,g}(x) + 1 = x_0x_3 + x_2x_3 + x_1 + x_2 + x_3 + 1. \end{aligned}$$

The previous observation can then be rewritten as $h_2 \circ S = h_3$. This pair of functions (h_2, h_3) therefore plays the same role as the pair (h_0, h_1) introduced by Beyne and presented in Section 4.2.3.a. It thus leads in the same way to a two-round invariant, but this time with $\left(\langle \texttt{0x2}, \texttt{0x5}\rangle^{16} \times \mathbb{F}_2^{64}\right) \cup \left(\mathbb{F}_2^{64} \times \langle \texttt{0x2}, \texttt{0x5}\rangle^{16}\right)$ as weak-key space. By observing that h_2 also appears among the components of $G_{a,g}$, namely $\texttt{0x6} \cdot G_{a,g} = h_2$, this implies that $S^{G_{a,g}}$ admits three exact linear approximations. Indeed, if we introduce $\alpha := \texttt{0xf}$, $\beta := \texttt{0x6}$ and $\gamma := \texttt{0x9}$, the following linear transitions hold with probability 1:

$$\alpha \xrightarrow{S^{G_{a,g}}} \beta, \quad , \beta \xrightarrow{S^{G_{a,g}}} \alpha, \quad \gamma \xrightarrow{S^{G_{a,g}}} \gamma,$$

the second one being a direct consequence of the involutive property of S.

This is reminiscent of our design choice to select conjugates that stay as close as possible from the ones of Beierle, Canteaut & Leander[BCL18]. However, besides the differences between weak-key spaces, the link between our differential distinguisher and the two-round invariants remains to be clarified. In particular, the previous example of a single linear trail with maximal correlation and the new example of a single differential trail with maximal probability that exist for conjugate ciphers is not only intriguing, it also points out how much such flaws can go unnoticed when only traditional linear and differential cryptanalysis are studied. More generally, it opens the question of whether an efficient unified framework could explain all these flaws at once, or equivalently, the question of whether all these attacks leverage the same weakness (that still needs to be precisely defined) only through different means.

4.4 Two other readings of the differential cryptanalysis of conjugates

The distinguisher of a differential nature that is presented above prompts further investigation. In this section, we start a theoretical analysis of this same phenomenon by presenting two equivalent points of view. The first one is a subcase of *commutative cryptanalysis* that is developed in [Bau+23] and that is presented in detail in the next chapter. We motivate its reason for being in the following. The second one was recently drawn to our attention at WCC 2024 by the work of Calderini, Civino & Invernizzi [CCI24a, CCI24b] which focuses on differential cryptanalysis using *other group laws* than the usual addition modulo 2. This section is then dedicated to drawing up a dictionary between the three points of view. In particular, our distinguisher, as well as two examples used in [CBS19, CCI24a, CCI24b] are further developed in Section 4.4.5.

4.4.1 Differential cryptanalysis of conjugates and commutative cryptanalysis

In the following, by the *cycle type* of a bijection $F\colon Z \to Z$ over a finite set Z, we mean the detail of how many cycles of each length are present in the cycle decomposition of F. Let us recall that conjugation is first and foremost, a set-theoretic notion that deals with the *cycle type* of a bijection, as it is highlighted by the following well-known result.

Proposition 4.4. *Let Z be a finite set. Let $F_0, F_1\colon Z \to Z$ be two permutations. Then F_0 and F_1 are conjugate if and only if F_0 and F_1 share the same cycle type. In other words, a conjugacy class is characterized by a cycle type.*

Proof. F_1 can be expressed as a composition of cyclic permutations with disjoint supports: $F_1 = \sigma_0 \circ \cdots \circ \sigma_{r-1}$. Let $G\colon Z \xrightarrow{\sim} Z$ be a bijection. Then, as already observed in Eq. (4.6), F_1^G can be decomposed as:

$$F_1^G = \sigma_0^G \circ \ldots \circ \sigma_{r-1}^G.$$

We therefore only look at $G \circ \sigma \circ G^{-1}$ for a cycle $\sigma = (a_0 \ \ldots \ a_{s-1})$. Let $x \in Z$. If $G^{-1}(x) = a_i$ for some i, then $G \circ \sigma \circ G^{-1}(x) = G(a_{i+1 \bmod s})$. In other words, $(G \circ \sigma \circ G^{-1})G(a_i) = G(a_{i+1 \bmod s})$ for any $i \in [\![0, s-1]\!]$. Otherwise, if $G^{-1}(x) \notin \{a_0, \ldots, a_{s-1}\}$, then $G \circ \sigma \circ G^{-1}(x) = G(G^{-1}(x)) = x$. This proves that σ^G is the cycle of length s (or s-cycle) $(G(a_0) \ \ldots \ G(a_{s-1}))$, and two conjugate permutations therefore share the same cycle type. Conversely, let us consider another s-cycle $\rho = (b_0 \ \ldots \ b_{s-1})$. Then any permutation G for which there exists j such that $\forall\, i, G(a_i) = b_{j+i \bmod s}$, gives a conjugacy relation between ρ and σ: $\rho^G = \sigma$. Thus, in the definition of such a G, only the image of $\mathrm{Supp}(\sigma)$ is constrained. By first pairing each cycle of F_1 to a cycle of F_2, and then building the look-up table of G using the partition $Z = \sqcup_{i\in[\![0,r-1]\!]}\mathrm{Supp}(\sigma_i)$, this proves that F_0 and F_1 are conjugate. □

The analysis of Section 4.3 then proves that the security of a cipher not only depends on its resistance to differential attacks, but also on the resistance of all bijections sharing the same cycle type.

This can also be understood as an *inherent* property of the original function $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ by using a generalized notion of *commutation*, rather than conjugation.

Definition 4.5 (Commutation)**.** Let $F, A, B\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$. Then, F and A commute if they satisfy $F \circ A = A \circ F$. By abuse of terminology, A and B are said to commute through F if they satisfy $F \circ A = B \circ F$. This situation is denoted by $A \xrightarrow{F} B$. More generally, the set of solutions that satisfy a commutation relation is denoted by:

$$Z_F^{\mathrm{comm}}(A, B) := \{x \in \mathbb{F}_2^n, F \circ A(x) = B \circ F(x)\}\,.$$

By another abuse, we say that A and B commute with probability $p \in [0, 1]$ through F if it holds that:

$$\frac{|Z_F^{\mathrm{comm}}(A, B)|}{2^n} = p.$$

Such a function A (resp. B) is called the input (resp. output) *commutant.* ▷

As hinted by the introduced notation, deterministic and probabilistic commutation relations through a cipher can be iteratively studied using *commutative trails*. This theory is developed in Chapter 5. For now, let us only consider its relationship with the differential cryptanalysis of conjugates with the following proposition. Recall that $Z_F^{\mathrm{diff}}(\Delta^{\mathrm{in}}, \Delta^{\mathrm{out}})$ is defined by:

$$Z_F^{\mathrm{diff}}(\Delta^{\mathrm{in}}, \Delta^{\mathrm{out}}) = \left\{x \in \mathbb{F}_2^n, F(x + \Delta^{\mathrm{in}}) + F(x) = \Delta^{\mathrm{out}}\right\}.$$

Proposition 4.6 (Conjugation, commutation and differentials)**.** *Let F, G be functions from $\mathbb{F}_2^n$ to itself where G is bijective. Let $\Delta^{\mathrm{in}}, \Delta^{\mathrm{out}} \in \mathbb{F}_2^n$. Then:*

1. $Z_F^{\mathrm{diff}}(\Delta^{\mathrm{in}}, \Delta^{\mathrm{out}}) = Z_F^{\mathrm{comm}}(T_{\Delta^{\mathrm{in}}}, T_{\Delta^{\mathrm{out}}})$, *and*
2. $Z_{F^G}^{\mathrm{diff}}(\Delta^{\mathrm{in}}, \Delta^{\mathrm{out}}) = G\left(Z_F^{\mathrm{comm}}\left(T_{\Delta^{\mathrm{in}}}^{G^{-1}}, T_{\Delta^{\mathrm{out}}}^{G^{-1}}\right)\right)$.

Proof. The first item is an immediate rewording (or a consequence of the second item with $G = \mathrm{Id}$). Regarding the second item we observe that:

$$\begin{aligned}
x \in Z_{F^G}^{\mathrm{diff}}(\Delta^{\mathrm{in}}, \Delta^{\mathrm{out}}) &\iff F^G(x) + F^G(x + \Delta^{\mathrm{in}}) = \Delta^{\mathrm{out}} \\
&\iff T_{\Delta^{\mathrm{out}}} G F G^{-1}(x) = G F G^{-1} T_{\Delta^{\mathrm{in}}}(x) \\
&\iff G^{-1} T_{\Delta^{\mathrm{out}}} G F G^{-1}(x) = F G^{-1} T_{\Delta^{\mathrm{in}}}(x) \\
&\iff G^{-1} T_{\Delta^{\mathrm{out}}} G F G^{-1}(x) = F G^{-1} T_{\Delta^{\mathrm{in}}} G G^{-1}(x) \\
&\iff T_{\Delta^{\mathrm{out}}}^{G^{-1}} \circ F(G^{-1}(x)) = F \circ T_{\Delta^{\mathrm{in}}}^{G^{-1}}(G^{-1}(x)) \\
&\iff G^{-1}(x) \in Z_F^{\mathrm{comm}}(T_{\Delta^{\mathrm{in}}}^{G^{-1}}, T_{\Delta^{\mathrm{out}}}^{G^{-1}}) \\
&\iff x \in G\left(Z_F^{\mathrm{comm}}(T_{\Delta^{\mathrm{in}}}^{G^{-1}}, T_{\Delta^{\mathrm{out}}}^{G^{-1}})\right)
\end{aligned}$$

□

As a direct reading of Proposition 4.6, we observe that differential cryptanalysis is a particular case of commutation, restricted to commutants that are translations. The differential study of F^G corresponds instead to commutation with commutants that are conjugates of translations, by G^{-1}, and not G. We denote by T the group of translations, *i.e.* $T := \{T_c, c \in \mathbb{F}_2^n\}$ and state the previous observation as the following informal corollary.

Corollary 4.7 (Conjugation and commutation)**.** *Let $F, G\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$, with G a bijection. Studying the differential properties of F^G is equivalent to studying the commutative properties of F with respect to commutants among the group $\left\{T_c^{G^{-1}}, c \in \mathbb{F}_2^n\right\} = G^{-1}TG$.*

This simple observation leads to multiple remarks. First of all, Corollary 4.7 is the differential counterpart of the work of Beierle, Canteaut & Leander [BCL18]. Indeed, commutative properties of a block cipher are related to differential properties of its conjugates, in the same way as non-linear approximations (and in particular invariants) are related to linear properties of the conjugates. Surprisingly however, these properties have never been studied before, at least in such terms and such generality.

However, Corollary 4.7 also points out that the considered class of commutants is really *restrictive*. Indeed, $G^{-1}TG$ is a conjugate of the group T and this implies that, like T, $G^{-1}TG$ is an Abelian 2-elementary regular group.

A group H is said to be *2-elementary* if each non-zero element is of order 2. A group $H \subset \mathrm{Bij}(\mathbb{F}_2^n)$ is said to be *regular* if it satisfies:

$$\forall (x, y) \in \mathbb{F}_2^n, \quad \exists!\ h \in H,\ h(x) = y.$$

The conjugation with T (or the regularity) in particular implies that any element of $G^{-1}TG$, except Id, is an involution without fixed point.

For these reasons, differential cryptanalysis of a conjugate cipher can only provide *exact* results about (either deterministic or probabilistic) commutations relations $A \xrightarrow{F} B$ where $A, B\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ are strongly constrained. For instance, it cannot exactly handle pairs (A, B) where one of the commutants is not a fixed-point-free involution (because of the 2-elementarity) or pairs for which there exists $x \in \mathbb{F}_2^n$, such that $A(x) = B(x)$ (because of the regularity). This is the reason why commutative cryptanalysis that is developed on its own in Chapter 5 provides *in theory* a larger class of attacks than the ones covered by differential analyzes of conjugates. Nonetheless, for an arbitrary pair (A, B), it is still possible to *approximate* the number of solutions of a given relation $A \xrightarrow{F} B$, by first *approximating* A and B by two elements of a given $G^{-1}TG$.

4.4.2 Elementary regular subgroups of the symmetric group

As already mentioned, another line of papers [CDVS05, CBS19, CCS21, CCI24a, CCI24b] tackles a similar problem from a group-theoretic perspective. In particular, Civino, Blondeau & Sala [CBS19] consider differential equations of a vectorial Boolean function $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ of the form:

$$F(x) \diamond F(x \diamond \Delta^{\text{in}}) = \Delta^{\text{out}},$$

where $\diamond\colon \mathbb{F}_2^n \times \mathbb{F}_2^n \to \mathbb{F}_2^n$ is an Abelian group operation for $\mathbb{F}_2^n$. This then generalizes the usual case which corresponds to the case where $\diamond$ is the bitwise addition.

Such an operation is built by the authors by first considering a regular 2-elementary Abelian subgroup $\mathcal{T} \subset \mathrm{Bij}(\mathbb{F}_2^n)$ of the symmetric group. In particular, given a regular group $\mathcal{T}$, we observe that $\{\tau(0), \tau \in \mathcal{T}\}$ is the full space $\mathbb{F}_2^n$. We can then enumerate $\mathcal{T}$ as $\mathcal{T} = \{\mathcal{T}_a, a \in \mathbb{F}_2^n\}$ where $\mathcal{T}_a$ is the unique function $\tau \in \mathcal{T}$ that satisfies $\tau(0) = a$.

Such a notation is chosen because a regular 2-elementary Abelian subgroup mimics the group of translations $T := \{T_a\colon x \mapsto x + a\}$ which is indeed made of fixed-point-free involutions (except $T_0 = \mathrm{Id}$), which commute one with the others, and where the only one that satifies $T_a(x) = y$ is T_{x+y}.

Let us clarify this mimicry. First, it is possible to build a group law $\diamond$ for which the group of translations is any regular 2-elementary (Abelian) subgroup of $\mathrm{Bij}(\mathbb{F}_2^n)$. The following proposition is the keystone of [CBS19].

Proposition 4.8 (Group law based on a regular subgroup)**.** *Let $\mathcal{T} \subset \mathrm{Bij}(\mathbb{F}_2^n)$ be a regular Abelian subgroup of the symmetric group. Let us define the operator $\diamond\colon \mathbb{F}_2^n \times \mathbb{F}_2^n \to \mathbb{F}_2^n$ by:*

$$\forall x, y \in \mathbb{F}_2^n, \quad x \diamond y := \mathcal{T}_x(y).$$

Then $(\mathbb{F}_2^n, \diamond)$ is an Abelian group. Furthermore $\mathcal{T}$ coincides with its group of translations.

Proof. Let us first observe that $\diamond$ is well-defined. It is a commutative operator because for any $x, y \in \mathbb{F}_2^n$ we have:

$$x \diamond y = \mathcal{T}_x(y) = \mathcal{T}_x(\mathcal{T}_y(0)) = \mathcal{T}_y(\mathcal{T}_x(0)) = \mathcal{T}_y(x) = y \diamond x.$$

Furthermore, for any $x \in \mathbb{F}_2^n$, we have by definition $x \diamond 0 = \mathcal{T}_x(0) = x$ so 0 is the identity element. As $\mathcal{T}_x^{-1} \in \mathcal{T}$, there exists y such that $\mathcal{T}_x^{-1} = \mathcal{T}_y$. We then observe that this element y satisfies:

$$x \diamond y = \mathcal{T}_x(y) = \mathcal{T}_x(\mathcal{T}_y(0)) = \mathcal{T}_x(\mathcal{T}_x^{-1}(0)) = 0,$$

which makes y the inverse of x. Finally for any $x, y, z \in \mathbb{F}_2^n$, we observe that:

$$x \diamond (y \diamond z) = \mathcal{T}_x(\mathcal{T}_y(z)), \quad \text{and} \quad (x \diamond y) \diamond z = \mathcal{T}_x(y) \diamond z = \mathcal{T}_{\mathcal{T}_x(y)}(z).$$

But $\mathcal{T}_{\mathcal{T}_x(y)}$ satisfies $\mathcal{T}_{\mathcal{T}_x(y)}(0) = \mathcal{T}_x(y)$ and $\mathcal{T}_x \circ \mathcal{T}_y$ belongs to $\mathcal{T}$ and also satisfies $\mathcal{T}_x \circ \mathcal{T}_y(0) = \mathcal{T}_x(y)$. By the regularity of $\mathcal{T}$, we necessarily have that $\mathcal{T}_{\mathcal{T}_x(y)} = \mathcal{T}_x \circ \mathcal{T}_y$ and therefore $\diamond$ is associative. So $(\mathbb{F}_2^n, \diamond)$ is indeed an Abelian group. Furthermore, for any $a \in \mathbb{F}_2^n$, the function $x \mapsto x \diamond a$ coincides by construction with $\mathcal{T}_a$. □

Remark 4.9. The previous proposition is stated without supposing that $\mathcal{T}$ is 2-elementary. If it is the case, then for any $x \in \mathbb{F}_2^n$, we have $\mathcal{T}_x^{-1} = \mathcal{T}_x$ and therefore x is its own inverse for the group law $\diamond$. In the following, this will always be the case. ▹

We can also go further in the parallel between such a group $\mathcal{T}$ and the group of translations thanks to the following well-known result.

Proposition 4.10. *Up to isomorphism, there exists a single 2-elementary group of order 2^n, which is $\mathbb{F}_2^n$.*

Proof. Any 2-elementary group H is actually Abelian since for any $x, y \in H$, $e = (xy)^2 = xyxy$. Therefore by left multiplication by x and right multiplication by y, we observe that for any $x, y \in H$ it holds that $xy = yx$.

Furthermore, there exists a unique definition for a scalar multiplication $\cdot \colon \mathbb{F}_2 \times H \to H$ as it should satisfy $1 \cdot h = h$ for any $h \in H$, but also $(1+1) \cdot h = (1 \cdot h)(1 \cdot h) = hh = e$, which implies that $0 \cdot h = e$ for any $h \in H$. This scalar multiplication satisfies all the necessary axioms making H a finite vector space over $\mathbb{F}_2$. Therefore, H is isomorphic to $\mathbb{F}_2^n$ as a vector space and *a fortiori* as a group. □

So any 2-elementary regular subgroup $\mathcal{T}$ of $\mathrm{Bij}(\mathbb{F}_2^n)$ is isomorphic to the group of translations T. But due to a result of Dixon [Dix71, proof of Lemma 1], we can be even more precise as two isomorphic regular subgroups of the symmetric group are necessarily conjugate.

Proposition 4.11 (Isomorphic and conjugate regular subgroups [Dix71])**.** *Let $n \geq 1$ and let S_n be the symmetric group of $[\![0, n-1]\!]$. Let H, K be two regular subgroups of S_n such that there exists a group isomorphism $\varphi \colon H \to K$. Then there exists $\sigma \in S_n$ such that $\sigma K \sigma^{-1} = H$.*

Proof. Adapted from [Dix71, Proof of Lemma 1]. Let us define the bijection $\sigma \colon [\![0, n-1]\!] \to [\![0, n-1]\!]$ by:

$$\forall h \in H, \quad \sigma(h(0)) := \varphi(h)(0). \tag{4.13}$$

The bijection σ is well-defined. Indeed, H is regular so $\{h(0), h \in H\} = [\![0, n-1]\!]$, but we also have $\{\varphi(h)(0), h \in H\} = \{k(0), k \in K\} = [\![0, n-1]\!]$ because φ is bijective and K is regular. By definition we also note that:

$$\forall h \in H, \quad h(0) = \sigma^{-1}(\varphi(h)(0)). \tag{4.14}$$

Let us enumerate K as $K = \{k_i, i \in [\![0, n-1]\!]\}$ where $k_i \in K$ is the unique $k \in K$ such that $k(0) = i$.

Let $h \in H$ and let us consider $\sigma \circ h \circ \sigma^{-1}$. Let $i \in [\![0, n-1]\!]$ and let us introduce $\widetilde{h} := \varphi(k_i)^{-1}$ and observe that by construction we have:

$$\varphi(\widetilde{h})(0) = \varphi(\varphi(k_i)^{-1})(0) = k_i(0) = i. \tag{4.15}$$

Then it holds that:

$$\begin{aligned}
\sigma \circ h \circ \sigma^{-1}(i) &= \sigma \circ h \circ \sigma^{-1}\left(\varphi\left(\widetilde{h}\right)(0)\right) && (4.16)\\
&= \sigma \circ h\left(\widetilde{h}(0)\right) && (4.17)\\
&= \sigma\left(h \circ \widetilde{h}(0)\right) && (4.18)\\
&= \varphi\left(h \circ \widetilde{h}\right)(0) && (4.19)\\
&= \varphi(h) \circ \varphi(\widetilde{h})(0) && (4.20)\\
&= \varphi(h)(i). && (4.21)
\end{aligned}$$

Eq. (4.16) comes from Eq. (4.15), Eq. (4.17) from Eq. (4.14), Eq. (4.18) is only a different bracket grouping, Eq. (4.19) comes from Eq. (4.13), Eq. (4.20) is due to the morphism property of φ and finally Eq. (4.21) is again due to Eq. (4.15).

All in all, it holds that $\sigma h \sigma^{-1} = \varphi(h)$, and this implies that $\sigma H \sigma^{-1} = K$. □

Let $\mathcal{T} = \{\mathcal{T}_a, a \in \mathbb{F}_2^n\}$ be a 2-elementary regular subgroup of $\mathrm{Bij}(\mathbb{F}_2^n)$. There therefore exists a bijection $G \colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ such that $G\mathcal{T}G^{-1} = T$. This also implies that there exists a bijection $\psi \colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ that satisfies:

$$\forall a \in \mathbb{F}_2^n, \quad G \circ \mathcal{T}_{\psi(a)} \circ G^{-1} = T_a. \tag{4.22}$$

By evaluating the previous equations at point $G(0)$, we obtain:

$$\forall a \in \mathbb{F}_2^n, \quad G \circ \mathcal{T}_{\psi(a)}(0) = G(0) + a, \quad \iff \quad G(\psi(a)) = G(0) + a.$$

In other words, for a given G such that $G\mathcal{T}G^{-1} = T$, there exists a single ψ satisfying Eq. (4.22) and it is defined by:

$$\forall a \in \mathbb{F}_2^n, \quad \psi(a) := G(G(0) + a)^{-1}.$$

Let $c, a \in \mathbb{F}_2^n$. The group $\mathcal{T}$ being Abelian, it holds that $\mathcal{T}_c \circ \mathcal{T}_{\psi(a)} = \mathcal{T}_{\psi(a)} \circ \mathcal{T}_c$, *i.e.*, $\mathcal{T}_c \circ \mathcal{T}_{\psi(a)} \circ \mathcal{T}_c^{-1} = \mathcal{T}_{\psi(a)}$. But as $\mathcal{T}$ is also 2-elementary, any element is its own inverse so $\mathcal{T}_c \circ \mathcal{T}_{\psi(a)} \circ \mathcal{T}_c = \mathcal{T}_{\psi(a)}$.

This implies that for any G, ψ satisfying Eq. (4.22), it also holds that for any $c \in \mathbb{F}_2^n$:

$$\forall a \in \mathbb{F}_2^n, \quad G \circ \mathcal{T}_c \circ \mathcal{T}_{\psi(a)} \circ \mathcal{T}_c \circ G^{-1} = T_a. \tag{4.23}$$

In other words, G can be replaced by $G \circ \mathcal{T}_c$ for any $c \in \mathbb{F}_2^n$. In particular, with $c = G^{-1}(0)$, we observe that $G \circ \mathcal{T}_{G^{-1}(0)}(0) = G(G^{-1}(0)) = 0$, so without loss of generality, we can always consider G such that $G(0) = 0$. In that case $\psi = G^{-1}$ and Eq. (4.22) can be simplified into:

$$\forall a \in \mathbb{F}_2^n, \quad G \circ \mathcal{T}_{G^{-1}(a)} \circ G^{-1} = T_a.$$

We restate this in the following proposition.

Proposition 4.12. *Let $\mathcal{T} = \{\mathcal{T}_a, a \in \mathbb{F}_2^n\}$ be a 2-elementary regular subgroup of* $\mathrm{Bij}(\mathbb{F}_2^n)$ *and $T = \{T_a \colon x \mapsto x + a, a \in \mathbb{F}_2^n\}$ be the group of translations for the usual addition law. Then there exists $G \in \mathrm{Bij}(\mathbb{F}_2^n)$ such that:*

$$\forall a \in \mathbb{F}_2^n, \quad G \circ \mathcal{T}_{G^{-1}(a)} \circ G^{-1} = T_a. \tag{4.24}$$

4.4.3 Differential cryptanalysis of conjugates and $\diamond$-differential cryptanalysis

We can henceforth translate the framework of [CBS19, CCI24a, CCI24b] into the differential cryptanalysis of some conjugates.

Remark 4.13. We stress that the relation between alternative group laws and conjugation is beyond any doubt well-known by the authors of [CDVS05, CBS19, CCS21, CCI24a, CCI24b] and mentioned multiple times in these papers. The novelty of the following formulation is that it relates this technique to commonly-used tools and notions of standard cryptanalysis. Furthermore, the following dictionary used in one way or the other provides more examples to each approach. ▷

Let $\diamond\colon \mathbb{F}_2^n \times \mathbb{F}_2^n \to \mathbb{F}_2^n$ be an Abelian group operation defined as in Proposition 4.8 for a 2-elementary regular subgroup $\mathcal{T} \subset \mathrm{Bij}(\mathbb{F}_2^n)$. As shown in the previous section, this is equivalent to saying that we only consider group laws $\diamond$ that are commutative and for which each element $x \in \mathbb{F}_2^n$ is its own inverse.

Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ be vectorial Boolean function and let us consider $\diamond$-differential equations of the form:

$$F(x) \diamond F(x \diamond \Delta^{\mathrm{in}}) = \Delta^{\mathrm{out}}, \tag{4.25}$$

for any $\Delta^{\mathrm{in}}, \Delta^{\mathrm{out}} \in \mathbb{F}_2^n$. In order to study Eq. (4.25), the notion of $\diamond$-differential probability is defined in [CBS19] for any ordered pair $(\Delta^{\mathrm{in}}, \Delta^{\mathrm{out}}) \in (\mathbb{F}_2^n)^2$ as the quantity denoted[3] by $\mathbb{P}\left[\Delta^{\mathrm{in}} \xrightarrow{F,\diamond} \Delta^{\mathrm{out}}\right]$ and defined by:

$$\mathbb{P}\left[\Delta^{\mathrm{in}} \xrightarrow{F,\diamond} \Delta^{\mathrm{out}}\right] := \frac{1}{2^n}\left|\left\{x \in \mathbb{F}_2^n, F(x) \diamond F(x \diamond \Delta^{\mathrm{in}}) = \Delta^{\mathrm{out}}\right\}\right|.$$

We also consider the associated set of solutions $Z_F^{\diamond\text{-diff}}(\Delta^{\mathrm{in}}, \Delta^{\mathrm{out}})$ that we define by:

$$Z_F^{\diamond\text{-diff}}(\Delta^{\mathrm{in}}, \Delta^{\mathrm{out}}) := \left\{x \in \mathbb{F}_2^n, F(x) \diamond F(x \diamond \Delta^{\mathrm{in}}) = \Delta^{\mathrm{out}}\right\}.$$

By Proposition 4.12, there exists $G\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ such that:

$$\forall a \in \mathbb{F}_2^n, \quad G \circ \mathcal{T}_{G^{-1}(a)} \circ G^{-1} = T_a.$$

The set $Z_F^{\diamond\text{-diff}}(\Delta^{\mathrm{in}}, \Delta^{\mathrm{out}})$ can then be equivalently defined by:

$$\begin{aligned}
Z_F^{\diamond\text{-diff}}(\Delta^{\mathrm{in}}, \Delta^{\mathrm{out}}) &:= \left\{x \in \mathbb{F}_2^n, F(x) \diamond F(x \diamond \Delta^{\mathrm{in}}) = \Delta^{\mathrm{out}}\right\} \\
&= \{x \in \mathbb{F}_2^n, \mathcal{T}_{\Delta^{\mathrm{out}}} \circ F(x) = F \circ \mathcal{T}_{\Delta^{\mathrm{in}}}(x)\} \\
&= \left\{x \in \mathbb{F}_2^n, G^{-1} \circ T_{G(\Delta^{\mathrm{out}})} \circ G \circ F(x) = F \circ G^{-1} \circ T_{G(\Delta^{\mathrm{in}})} \circ G(x)\right\} \\
&= Z_F^{\mathrm{comm}}\left(T_{G(\Delta^{\mathrm{in}})}^{G^{-1}}, T_{G(\Delta^{\mathrm{out}})}^{G^{-1}}\right).
\end{aligned}$$

Combined with Proposition 4.6, we obtain the following proposition.

[3] In [CBS19], this quantity is denoted by $p^{\diamond}_{(\Delta^{\mathrm{in}},\Delta^{\mathrm{out}}),F}$. We choose our notation for uniformity along this manuscript.

Theorem 4.14 ($\diamond$-differential, conjugation & commutation)**.** *Let $(\mathbb{F}_2^n, \diamond)$ be an Abelian group such that $\mathcal{T} := \{x \mapsto x \diamond c, \forall c \in \mathbb{F}_2^n\}$ is 2-elementary and regular. Let $G\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ be such that: $\forall a \in \mathbb{F}_2^n, \quad G \circ \mathcal{T}_{G^{-1}(a)} \circ G^{-1} = T_a$. Then, it holds that:*

$$Z_F^{\diamond\text{-}diff}(\Delta^{\text{in}}, \Delta^{\text{out}}) = Z_F^{\text{comm}}\left(T^{G^{-1}}_{G(\Delta^{\text{in}})}, T^{G^{-1}}_{G(\Delta^{\text{out}})}\right) = G^{-1}\left(Z^{\text{diff}}_{F^G}\left(G(\Delta^{\text{in}}), G(\Delta^{\text{out}})\right)\right). \tag{4.26}$$

In other words, Theorem 4.14 states that studying the $\diamond$-differential properties of F is equivalent to either studying the differential properties of its conjugate F^G or the commutation with commutants among the group $T^{G^{-1}}$.

Stated otherwise, the two methodologies from [CBS19, CCI24a, CCI24b] and from [Bau+23] coincide: despite the clear difference of flavors, they both study the differential properties of a conjugate cipher $E^G = G \circ F_k^{(R-1)} \circ \ldots \circ F_k^{(0)} \circ G^{-1}$ by leveraging weaknesses of the conjugate round functions $(F_k^{(r)})^G$ for any r. We also note that both approaches benefit from the study of the other.

In order to use this dictionary in both ways, we clarify that such a change of variables G is in practice easy to build.

Lemma 4.15 (Characterization of G)**.** *Let $(\mathbb{F}_2^n, \diamond)$ be an Abelian group such that $\mathcal{T} := \{x \mapsto x \diamond c, \forall c \in \mathbb{F}_2^n\}$ is 2-elementary and regular. Let $G\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$. Then G satisfies $\forall a \in \mathbb{F}_2^n, \quad G \circ \mathcal{T}_{G^{-1}(a)} \circ G^{-1} = T_a$ if and only if G is a group isomorphism from $(\mathbb{F}_2^n, \diamond)$ to $(\mathbb{F}_2^n, +)$.*

Proof. The mapping G satisfies the first condition if and only if it holds that:

$$\forall\, x, y \in \mathbb{F}_2^n, \quad x \diamond y = \mathcal{T}_y(x) = G^{-1} \circ T_{G(y)} \circ G(x).$$

This is naturally equivalent to:

$$\forall\, x, y \in \mathbb{F}_2^n, G(x \diamond y) = T_{G(y)}(G(x)) = G(x) + G(y).$$

□

As noted in the proof of Proposition 4.10, $(\mathbb{F}_2^n, \diamond)$ is actually a vector space over $\mathbb{F}_2^n$. We can then fix a basis $(b_0, \ldots, b_{n-1})$ such that any element $x \in \mathbb{F}_2^n$ can be uniquely decomposed as $x = y_0 b_0 \diamond y_1 b_1 \diamond \ldots \diamond y_{n-1} b_{n-1}$, with $y_i \in \mathbb{F}_2$ for all i. By Lemma 4.15, G must then satisfy:

$$\forall x \in \mathbb{F}_2^n, \quad G(x) = \sum_{i=0}^{n-1} y_i G(b_i).$$

Building such a G is therefore equivalent to selecting a basis $(B_0, \ldots, B_{n-1})$ of $(\mathbb{F}_2^n, +)$, defining $G(b_i) = B_i$ for any i, and expanding the definition by “linearity”.

4.4.4 Discussions on the weak-key space

With the link drawn in the previous subsections, we can compare the approach of Section 4.3 with the one of [CBS19, CCI24a, CCI24b], and especially the introduced notions of weak keys.

4.4.4.a Weak-key space of [CBS19]

The authors of [CBS19] introduced a weak-key space that is denoted by $W^\diamond$ and defined by:

$$W^\diamond := \{k \in \mathbb{F}_2^n,\ T_k = \mathcal{T}_k\}\,.$$

Remark 4.16. The set $W^\diamond$ is defined by $W^\diamond = \{k \in \mathbb{F}_2^n, T_k \in \mathcal{T}\}$ in [CCS21]. Both definitions coincide because $\mathcal{T}$ is regular, so the condition $T_k \in \mathcal{T}$ necessarily implies that T_k and $\mathcal{T}_k$ must coincide because $T_k(0) = k = \mathcal{T}_k(0)$. ▷

From the conjugate point-of-view, $W^\diamond$ can be described as:

$$\begin{aligned} W^\diamond &= \{k \in \mathbb{F}_2^n, T_k = \mathcal{T}_k\} \\ &= \left\{k \in \mathbb{F}_2^n, T_k = G^{-1} \circ T_{G(k)} \circ G\right\} \\ &= \left\{k \in \mathbb{F}_2^n, T_k^G = T_{G(k)}\right\}. \end{aligned}$$

In other words, $W^\diamond$ is the set of k such that the conjugate T_k^G is still a constant addition, with a possibly different constant. But, because of Lemma 2.27, a function $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ is affine with L as linear part if and only if it satisfies:

$$\forall \Delta^{\text{in}}, \Delta^{\text{out}} \in \mathbb{F}_2^n,\ \mathbb{P}\left[\Delta^{\text{in}} \xrightarrow{F} \Delta^{\text{out}}\right] = \begin{cases} 1 & \text{if } \Delta^{\text{in}} = L(\Delta^{\text{out}}) \\ 0 & \quad\text{otherwise} \end{cases},$$

We can thus redefine $W^\diamond$ as in the following lemma.

Lemma 4.17 ($W^\diamond$ as a weak-key space)**.** *Let* $(\mathbb{F}_2^n, \diamond)$ *be an Abelian group such that* $\mathcal{T} := \{x \mapsto x \diamond c, \forall c \in \mathbb{F}_2^n\}$ *is 2-elementary and regular. Let* $G\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ *be such that:* $\forall a \in \mathbb{F}_2^n,\quad G \circ \mathcal{T}_{G^{-1}(a)} \circ G^{-1} = T_a$. *Then,*

$$W^\diamond = \left\{k \in \mathbb{F}_2^n,\ \forall\ \Delta^{\text{in}}, \Delta^{\text{out}} \in \mathbb{F}_2^n,\quad \mathbb{P}\left[\Delta^{\text{in}} \xrightarrow{T_k^G} \Delta^{\text{out}}\right] = \mathbf{1}_{\Delta^{\text{in}}}(\Delta^{\text{out}})\right\},$$

where $\mathbf{1}_x(y) = 1$ *if* $x = y$ *and* 0 *otherwise.*

The description of $W^\diamond$ drawn in Lemma 4.17 explains the fact that it is indeed a weak-key space: whenever, k belongs to $W^\diamond$, any differential transition through T_k^G is deterministic. Stated otherwise, such a transition only depends on the differences and is independent of the actual *values* of the considered pairs.

While Lemma 4.17 clearly outlines the importance of the set $W^\diamond$ in such a study, its structure can be further clarified. This is the purpose of Lemma 4.19, which relies on the notion of *linear structures*.

Definition 4.18 (Linear structure)**.** Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$. Let $\Delta \in \mathbb{F}_2^n$. The difference Δ is said to be a *linear structure* of F if the derivative $D_\Delta F$ is a constant function. The set of linear structures of F is denoted by $\mathrm{LS}\,(F)$, that is:

$$\mathrm{LS}\,(F) := \{\Delta \in \mathbb{F}_2^n,\ \forall\, x \in \mathbb{F}_2^n, D_\Delta F(x) = F(0) + F(\Delta)\}\,.$$

▷

Lemma 4.19 ($W^\diamond$ as linear space of G)**.** *Let $(\mathbb{F}_2^n, \diamond)$ be an Abelian group such that $\mathcal{T} := \{x \mapsto x \diamond c, \forall c \in \mathbb{F}_2^n\}$ is 2-elementary and regular. Let $G\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ be such that: $\forall a \in \mathbb{F}_2^n$, $\quad G \circ \mathcal{T}_{G^{-1}(a)} \circ G^{-1} = T_a$. Then, $W^\diamond = \mathrm{LS}\,(G)$.*

Proof. Starting from the first definition of $W^\diamond$, we observe that:

$$\begin{aligned}
W^\diamond &= \{k \in \mathbb{F}_2^n, T_k = \mathcal{T}_k\} \\
&= \left\{k \in \mathbb{F}_2^n, T_k = G^{-1} \circ T_{G(k)} \circ G\right\} \\
&= \left\{k \in \mathbb{F}_2^n, G \circ T_k = T_{G(k)} \circ G\right\} \\
&= \left\{k \in \mathbb{F}_2^n, \left|Z_G^{\mathrm{comm}}(T_k, T_{G(k)})\right| = 2^n\right\} \\
&= \left\{k \in \mathbb{F}_2^n, \left|Z_G^{\mathrm{diff}}\,(k, G\,(k))\right| = 2^n\right\} \\
&= \mathrm{LS}\,(G)\,,
\end{aligned}$$

where we use the first item of Proposition 4.6 for the fifth equality. The last equality holds because the value of a constant derivative $D_k G$ is necessarily $G(k) + G(0)$ but we have by construction that $G(0) = 0$. □

In light of the following results, the notion of linear structure is relatively well-understood and gives new insights on this set of weak keys.

Lemma 4.20 (Standard properties of linear structures)**.** *Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$. Then:*

1. *$\mathrm{LS}\,(F)$ is a linear space and the restriction of F to $\mathrm{LS}\,(F)$ is affine.*

2. *If F is bijective then $\mathrm{LS}\,(F^{-1}) = F(0) + F(\mathrm{LS}\,(F))$ and in particular $\dim(\mathrm{LS}\,(F)) = \dim(\mathrm{LS}\,(F^{-1}))$.*

3. *[Lai95, Theorem 3] Let $r := \dim(\mathrm{LS}\,(F))$. Then F is linearly equivalent to a function $G\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ defined by:*

$$G\colon (x_0, \ldots, x_{n-1}) \mapsto L(x_0, \ldots, x_{r-1}) + \widetilde{F}(x_r, \ldots, x_{n-1}),$$

where $L\colon \mathbb{F}_2^r \to \mathbb{F}_2^n$ is linear and $\widetilde{F}\colon \mathbb{F}_2^{n-r} \to \mathbb{F}_2^n$ satisfies $\mathrm{LS}\left(\widetilde{F}\right) = \{0\}$.

The proofs of the previous statements are given for the sake of completeness.

Proof. Adapted from [Lai95, Theorem 3]. Let $x, y \in \mathrm{LS}(F)$. Let $z \in \mathbb{F}_2^n$. Then:

$$\begin{aligned}
F(z+x+y)+F(x+y) &= F(z+x)+F(y)+F(0)+F(x+y)\\
&= F(z)+F(x)+F(y)+F(x+y)\\
&= F(z)+F(x)+F(y)+F(x)+F(y)+F(0)\\
&= F(z)+F(0),
\end{aligned}$$

where we successively use, the facts that $y \in \mathrm{LS}(F)$, then $x \in \mathrm{LS}(F)$, and again $y \in \mathrm{LS}(F)$. In other words, $x+y$ belongs to $\mathrm{LS}(F)$, so we deduce that $\mathrm{LS}(F)$ is a linear space. Furthermore we observe that for any $x, y \in \mathrm{LS}(F)$, we have $F(x+y) = F(x)+F(y)+F(0)$ (by using the fact that either x or y belongs to $\mathrm{LS}(F)$), which means that F is indeed affine on $\mathrm{LS}(F)$.

Regarding the second item, $\Delta^{\mathrm{in}} \in \mathbb{F}_2^n$ belongs to $\mathrm{LS}(F)$ if and only if $F \circ T_{\Delta^{\mathrm{in}}} = T_{\Delta^{\mathrm{out}}} \circ F$ where $\Delta^{\mathrm{out}} = F(0)+F(\Delta^{\mathrm{in}})$. It is therefore equivalent to $F^{-1} \circ T_{\Delta^{\mathrm{out}}} = T_{\Delta^{\mathrm{in}}} \circ F^{-1}$ and therefore to the fact that $\Delta^{\mathrm{out}} \in \mathrm{LS}\left(F^{-1}\right)$ and thus $F(0)+F(\mathrm{LS}(F)) \subset \mathrm{LS}\left(F^{-1}\right)$. For the same reason, we have $F^{-1}(0)+F^{-1}(\mathrm{LS}\left(F^{-1}\right)) \subset \mathrm{LS}(F)$. This implies that $|\mathrm{LS}(F)| = \left|\mathrm{LS}\left(F^{-1}\right)\right|$, the inclusion $F(0)+F(\mathrm{LS}(F)) \subset \mathrm{LS}\left(F^{-1}\right)$ is then an equality.

Let us now consider a basis $(a_0, \ldots, a_{r-1})$ of $\mathrm{LS}(F)$ and complete it into a basis of $\mathbb{F}_2^n$ that we denote by $(a_0, \ldots, a_{n-1})$. Let A be the linear bijection defined by $A(\xi^{(i)}) = a_i$ for any $i \in [\![0, n-1]\!]$. Let $x = (x_0, \ldots, x_{n-1}) \in \mathbb{F}_2^n$. Then it holds that:

$$\begin{aligned}
F \circ A(x_0, \ldots, x_{n-1}) &= F \circ A\left(\sum_{i=0}^{n-1} x_i \xi^{(i)}\right)\\
&= F\left(\sum_{i=0}^{n-1} x_i a_i\right)\\
&= F\left(\sum_{i=0}^{r-1} x_i a_i\right) + F\left(\sum_{j=r}^{n-1} x_j a_j\right) + F(0)\\
&= \sum_{i=0}^{r-1} x_i(F(a_i)+F(0)) + F\left(\sum_{j=r}^{n-1} x_j a_j\right),
\end{aligned}$$

where we successively use the linearity of A, the fact that $\sum_{i=0}^{r-1} x_i a_i$ is a linear structure of F, and the fact that $F + F(0)$ is linear over $\mathrm{LS}(F)$. Let us define L and $\widetilde{F}$ by:

$$\begin{aligned}
L &\colon (y_0, \ldots, y_{r-1}) \mapsto \sum_{i=0}^{r-1} y_i(F(a_i)+F(0)), \text{ and}\\
\widetilde{F} &\colon (y_0, \ldots, y_{n-r-1}) \mapsto F\left(\sum_{j=0}^{n-r-1} y_j a_{j+r}\right).
\end{aligned}$$

The function L is by construction linear. Let $c = (c_0, \dots, c_{n-r-1})$ be a linear structure of $\widetilde{F}$. Let $b = (0, \dots, 0, c_0, \dots, c_{n-r-1}) \in \mathbb{F}_2^n$. Then, for any $x \in \mathbb{F}_2^n$:

$$\begin{aligned} F \circ A(x+b) &= L(x_0, \dots x_{r-1}) + \widetilde{F}(x_r + b_r, \dots, x_{n-1} + b_{n-1}) \\ &= L(x_0, \dots x_{r-1}) + \widetilde{F}(x_r, \dots, x_{n-1}) + \widetilde{F}(c) + \widetilde{F}(0) \\ &= F \circ A(x) + F \circ A(b) + F(0), \end{aligned}$$

or stated otherwise $F(Ax + Ab) = F(Ax) + F(Ab) + F(0)$. But A being bijective this implies that Ab is a linear structure of F. However, as $Ab = \sum_{i=r}^{n-1} a_i b_i$ with $a_i \notin \mathrm{LS}(F)$ for any $i \in [\![r, n-1]\!]$, this implies that $b_i = 0$ for any such i, and therefore $c = 0$. The function $\widetilde{F}$ thus only has a trivial linear structure. □

Corollary 4.21 (Dimension of $\mathrm{LS}(F)$). *Let $F \colon \mathbb{F}_2^n \to \mathbb{F}_2^n$. If F is non-linear then* $\dim(\mathrm{LS}(F)) \leq n-2$. *Furthermore, for any F,* $\deg_a(F) \leq n - \dim(\mathrm{LS}(F))$. *In particular, if* $\dim(\mathrm{LS}(F)) = n-2$, *then* $\deg_a(F) = 2$.

Proof. Let us suppose that F is non-linear and that $\dim(\mathrm{LS}(F)) \in \{n-1, n\}$. If $\dim(\mathrm{LS}(F)) = n$, then $\mathrm{LS}(F)$ is the full space, and by the first item of Lemma 4.20, F is affine over $\mathbb{F}_2^n$, which contradicts the non-linearity hypothesis. Therefore, we have $\dim(\mathrm{LS}(F)) = n-1$. But in that case the third item of Lemma 4.20 implies that F is linearly equivalent to a function $G \colon x \mapsto L(x_0, \dots, x_{n-2}) + \widetilde{F}(x_{n-1})$ such that L is linear. But the function of a single variable $\widetilde{F}$ is either constant or affine. This implies that F is linearly equivalent to an affine function, which again contradicts the non-linearity of F. The fact that for any F, it holds that $\deg_a(F) \leq n - \dim(\mathrm{LS}(F))$ is also a consequence of the third item: F is indeed equivalent to a function which can only be non-linear in its $n - \dim(\mathrm{LS}(F))$ last variables. □

Upper bound on $W^\diamond = \mathrm{LS}(G)$. This good understanding of linear structures can then be applied to our case. Recall that in order to be an interesting change of variables, G must be non-linear. In light of Corollary 4.21, it then satisfies $\dim(\mathrm{LS}(G)) \leq n-2$. In this particular context, this provides the following upper bound on the number of weak keys for this attack:

$$\dim(W^\diamond) \leq n-2.$$

This result, which is stated in [CCS21, Proposition 4.1], can then be seen as a consequence of the more general result about linear structures.

The case $\dim(W^\diamond) = \dim(\mathrm{LS}\,(G)) = n-2$**.** By Corollary 4.21, the maximum number of weak-keys for the attack of Civino, Blondeau & Sala [CBS19] corresponds in our context to some specific quadratic G. More precisely, let G be a bijection such that $\dim(\mathrm{LS}\,(G)) = n-2$. Because of the third item of Lemma 4.20, G can be written as $G = H \circ A^{-1}$, where A is a linear bijection and $H\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ a function whose components are made only of constant terms, affine terms and the sole quadratic term x_0x_1. Note that x_0x_1 must appear in at least one coordinate, but not necessarily all of them. In particular, there exists a bijective linear mapping B such that $B \circ H$ has a single coordinate containing x_0x_1 while all others are affine. However, the differential properties of F^G and the ones of $F^{B\circ G}$ are identical as $F^{B\circ G} = B \circ F^G \circ B^{-1}$. This implies that the choices of change of variables G for which the number of weak keys is maximal are similar to the choices we previously made in Section 4.3.1.b. This in particular applies to the choices made by the authors of [CBS19, CCI24a, CCI24b] which focused on maximizing the number of weak-keys.

Note that, because of Lemma 4.20, the specific case $\dim(\mathrm{LS}\,(G)) = n-2$ implies that $\dim(\mathrm{LS}\,(G^{-1})) = 2$, and due to Corollary 4.21, both G and G^{-1} are in that case quadratic.

Lower bound on $W^\diamond = \mathrm{LS}\,(G)$**.** Civino, Blondeau & Sala [CBS19] actually choose G with at least one non-trivial linear structure. This is guaranteed whenever $\mathcal{T}$ is a subgroup of the affine group. This is presented in the following proposition which is a particular case of a result due to Caranti, Dalla Volta & Sala [CDVS05].

Proposition 4.22 (Non-trivial weak-key space [CDVS05])**.** *Let* $\mathcal{T}$ *be a 2-elementary regular subgroup of the affine group* $\mathrm{Aff}(\mathbb{F}_2^n)$*. Then* $\mathcal{T} \cap T \neq \{\mathrm{Id}\}$*, and thus* $W^\diamond = \{k \in \mathbb{F}_2^n, T_k \in \mathcal{T} \cap T\} \neq \{0\}$*.*

Proof. Adapted from [CDVS05]. For any x, because $\mathcal{T}_x$ is affine and satisfies $\mathcal{T}_x(0) = x = T_x(0)$, it can be decomposed as $\mathcal{T}_x = T_x \circ L_x$ where L_x is linear. Because $\mathcal{T}_x^2 = \mathrm{Id}$, it holds for any $z \in \mathbb{F}_2^n$ that:

$$z = L_x(L_x(z) + x) + x = L_x^2(z) + L_x(x) + x.$$

In particular with $z = 0$, we observe that $L_x(x) = x$. Therefore, we also get $L_x^2 = \mathrm{Id}$. Let $x, y \in \mathbb{F}_2^n$. Then:

$$\begin{aligned}
\mathcal{T}_x T_y \mathcal{T}_x &= T_x L_x T_y T_x L_x \\
&= T_x L_x T_{y+x} L_x \\
&= T_x T_{L_x(y+x)} L_x L_x \\
&= T_{x+L_x(y+x)} \\
&= T_{L_x(x)+L_x(y+x)} \\
&= T_{L_x(y)},
\end{aligned} \tag{4.27}$$

where we successively use the decomposition of $\mathcal{T}_x$, the fact that $T_y T_x = T_{y+x}$, the fact that $L_x T_{y+x} = T_{L_x(y+x)} L_x$ because L_x is linear, then $L_x^2 = \mathrm{Id}$ and finally

$x = L_x(x)$ and the linearity of L_x again. In particular, we observe that $\mathcal{T}_x T_y \mathcal{T}_x \in T$ for any $x, y \in \mathbb{F}_2^n$. This implies that the function $F\colon \mathcal{T} \times T \to T$ that is defined by:

$$F\colon (\mathcal{T}_x, T_y) \mapsto \mathcal{T}_x T_y \mathcal{T}_x,$$

is actually well-defined. It corresponds to the action by conjugation of $\mathcal{T}$ on the set T as $\mathcal{T}_x^{-1} = \mathcal{T}_x$ for any x in our case. As for any action of a p-group H on a set Z, the orbit-stabilizer theorem states that the number of elements that are H-invariants is equal to $|Z|$ modulo p. In our case, the set Z of $\mathcal{T}$-invariants is defined by:

$$Z := \{T_y, \mathcal{T}_x T_y \mathcal{T}_x = T_y \forall x \in \mathbb{F}_2^n\}$$

and it must be of even cardinality. As it contains $T_0 = \mathrm{Id}$, it must contain at least a non-trivial element. To conclude, we now show that Z is actually equal to $\mathcal{T} \cap T$. Indeed, we have:

$$\begin{aligned} Z &= \left\{T_y, T_{L_x(y)} = T_y,\ \forall\, x \in \mathbb{F}_2^n\right\} \\ &= \{T_y, L_x(y) + y = 0,\ \forall\, x \in \mathbb{F}_2^n\} \\ &= \{T_y, L_y(x) + x = 0,\ \forall\, x \in \mathbb{F}_2^n\} \\ &= \{T_y, L_y = \mathrm{Id}\} \\ &= \{T_y, \mathcal{T}_y = T_y\} \\ &= \mathcal{T} \cap T. \end{aligned}$$

The third equality holds because for any $x, y \in \mathbb{F}_2^n$, we have:

$$L_x(y) + y = \mathcal{T}_x(y) + y + x = \mathcal{T}_y(x) + y + x = L_y(x) + x.$$

□

Corollary 4.23. *Let $G\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ be bijective. Let us suppose that $G \circ T_c \circ G^{-1}$ is affine for any $c \in \mathbb{F}_2^n$. Then* $\dim(\mathrm{LS}\,(G)) \geq 1$.

4.4.4.b Comparison with our weak-key space $W(\Delta, \Delta)$

As shown in Lemma 4.17, whenever a key belongs to $W^\diamond$, the actual key does not matter anymore as the behavior is deterministic and independent of the key. This enables us to launch *any kind of differential attacks* as it is done in the classical way.

However, contrary to the standard case, the fraction of the keys cannot exceed one quarter of the key space. In the setting where the change of variable is a parallel application of a non-linear change of variables of the size of the Sbox, this fraction is in practice way smaller.

The set $W^\diamond$ is actually a conservative choice of weak-key space in the sense that it is built so that *any* differential attack works. On the contrary, if we are instead interested in a *specific* attack taking advantage of some specific transition, we can hope for a bigger set of weak keys. Let $\Delta^{\mathrm{in}}, \Delta^{\mathrm{out}} \in \mathbb{F}_2^n$, and let us consider

$\Delta^{\text{in}} \xrightarrow{T_k^G} \Delta^{\text{out}}$. In that case, we actually want to consider the set $W(\Delta^{\text{in}}, \Delta^{\text{out}})$ that is defined by:

$$\begin{aligned} W(\Delta^{\text{in}}, \Delta^{\text{out}}) &:= \left\{ k \in \mathbb{F}_2^n, \mathbb{P}\left[\Delta^{\text{in}} \xrightarrow{T_k^G} \Delta^{\text{out}}\right] = 1 \right\} \\ &= \left\{ k \in \mathbb{F}_2^n, \forall\ x \in \mathbb{F}_2^n, \quad D_{\Delta^{\text{in}}} T_k^G(x) = \Delta^{\text{out}} \right\} \end{aligned}$$

In other words, we would like to consider linear structures shared by multiples T_k^G, for which the corresponding constant derivatives are equal. This definition actually generalizes the set introduced in Eq. (4.10) on page 130. A direct corollary of Lemma 4.17 is that for a given $\Delta \in \mathbb{F}_2^n$ we have:

$$\text{LS}\,(G) \subset W(\Delta, \Delta).$$

However in practice, the set $\text{LS}\,(G)$ can be a strict subset of $W(\Delta, \Delta)$. An example is given in Section 4.4.5 below. Furthermore, while transitions $\Delta \xrightarrow{T_k^G} \Delta$ with probability 1 imitates the standard differential case for the bitwise addition, there might also exist $\Delta^{\text{in}} \neq \Delta^{\text{out}}$ such that $W(\Delta^{\text{in}}, \Delta^{\text{out}}) \neq \emptyset$. Therefore, transitions $\Delta^{\text{in}} \xrightarrow{T_k^G} \Delta^{\text{out}}$ with probability 1 can also be considered in a weak-key setting. This is in particular important in the case where T_k^G is affine for any k. In that specific case, $W(\Delta^{\text{in}}, \Delta^{\text{out}})$ becomes:

$$W(\Delta^{\text{in}}, \Delta^{\text{out}}) = \left\{ k \in \mathbb{F}_2^n, D_{\Delta^{\text{in}}} T_k^G(0) = \Delta^{\text{out}} \right\},$$

because any derivative of any T_k^G is constant. This also means that for a fixed $\Delta^{\text{in}} \in \mathbb{F}_2^n$, the sets $W(\Delta^{\text{in}}, \Delta^{\text{out}})$ for all $\Delta^{\text{out}} \in \mathbb{F}_2^n$ partition the set of round keys:

$$\bigsqcup_{\Delta^{\text{out}} \in \mathbb{F}_2^n} W(\Delta^{\text{in}}, \Delta^{\text{out}}) = \{k \in \mathbb{F}_2^n\} = \mathbb{F}_2^n.$$

4.4.5 Complementing some examples

4.4.5.a The previous conjugate of Midori

Let us take a look back at the analysis of the specific conjugate of Midori that is addressed in Section 4.3. In particular, let us denote by $G\colon \mathbb{F}_2^4 \to \mathbb{F}_2^4$ the function $G_{a,g}$ given in Table 4.6. Its ANF is given by:

$$G(x) = \begin{pmatrix} x_0x_3 + x_0 + x_2x_3 \\ x_0 + x_2 \\ x_1 \\ x_3 \end{pmatrix}. \tag{4.28}$$

Because, we are interested in the properties of S^G, Theorem 4.14 states that we can equivalently study the $\diamond$-differential properties for a specific law $\diamond$. The group of translations of this law $\diamond$ is $\mathcal{T} = G^{-1}TG$. For this reason, we can look at the conjugates $T_k^{G^{-1}}$. Their ANFs, as well as the ANF of $\diamond$ are given by:

$$\forall x, k \in \mathbb{F}_2^4, \quad x \diamond k := T_{G(k)}^{G^{-1}}(x) = \begin{pmatrix} x_0 + k_0 + (x_0 + x_2)k_3 + x_3(k_0 + k_2) \\ x_1 + k_1 \\ x_2 + k_2 + (x_0 + x_2)k_3 + x_3(k_0 + k_2) \\ x_3 + k_3 \end{pmatrix}. \tag{4.29}$$

From this ANF, we can easily observe that:

$$W^\diamond = \left\{k \in \mathbb{F}_2^4, T_k = \mathcal{T}_k\right\} = \left\{k \in \mathbb{F}_2^4, k_3 = 0, k_0 = k_2\right\} = \langle \texttt{0x2}, \texttt{0x5} \rangle .$$

Because of Lemma 4.19, we can determine $W^\diamond$ without this explicit formula for $\diamond$. Indeed, it suffices to look at the linear structures of G. From the ANF of G given in Eq. (4.28), it is clear that:

$$\forall x, \Delta \in \mathbb{F}_2^4, \quad D_\Delta G(x) = \begin{pmatrix} \Delta_0 + x_3(\Delta_0 + \Delta_2) + \Delta_3(x_0 + x_2) + \Delta_3(\Delta_0 + \Delta_2) \\ \Delta_0 + \Delta_2 \\ \Delta_1 \\ \Delta_3 \end{pmatrix}.$$

The derivative $D_\Delta G$ is therefore constant if and only if $\Delta_0 = \Delta_2$ and $\Delta_3 = 0$, and, as expected, the same set $W^\diamond$ is obtained.

However, recall that while focusing on the specific transition $\Delta \xrightarrow{T_k^G} \Delta$ where $\Delta = \texttt{0xd} = \texttt{0b1101}$, we computed in Eq. (4.12) (and below) that:

$$W(\Delta, \Delta) = \left\{k \in \mathbb{F}_2^4, k_0 + k_2 = 0\right\} = \langle \texttt{0x2}, \texttt{0x5}, \texttt{0x8} \rangle .$$

In particular, $W(\Delta, \Delta)$ strictly contains $W^\diamond$ and the differential trail $\nabla \xrightarrow{(F_k^{(0)})^{\mathcal{G}}} \nabla \to \cdots \xrightarrow{(F_k^{(R-1)})^{\mathcal{G}}} \nabla$ holds with probability 1 if all nibbles of all rounds keys and round constants belong to $W(\Delta, \Delta)$.

Recall that this trail is based on the differential transition $\Delta \xrightarrow{S^G} \Delta$ that holds with probability 1. Because of Theorem 4.14, this property can equivalently be considered as the probability-1 $\diamond$-differential transition $G^{-1}(\Delta) \xrightarrow{S,\diamond} G^{-1}(\Delta)$ for the law $\diamond$ given above, or as probability-1 commutation with the affine function $A := T_\Delta^{G^{-1}}$ where $\Delta = \texttt{0xd}$. The ANF of A can easily be deduced from Eq. (4.29) using $k = G^{-1}(\Delta) = \texttt{0xf} = \texttt{0b1111}$ and is given below:

$$A(x) := \begin{pmatrix} x_2 + 1 \\ x_1 + 1 \\ x_0 + 1 \\ x_3 + 1 \end{pmatrix}. \tag{4.30}$$

4.4.5.b The toy ciphers of [CBS19, CCI24a, CCI24b]

The toy cipher of [CBS19]. In [CBS19], a block cipher with a 15-bit state is proposed to illustrate $\diamond$-differential cryptanalysis. It has a standard SPN structure where the Sbox layer is the 5-time parallel application of a single 3-bit Sbox $S\colon \mathbb{F}_2^3 \to \mathbb{F}_2^3$.

The look-up table of S is given in Table 4.7.

x	0	1	2	3	4	5	6	7
$S(x)$	0	6	2	1	5	7	4	3

x	0	1	2	3	4	5	6	7
$G(x)$	0	1	2	3	6	7	5	4

Table 4.7: The Sbox used in [CBS19] and a suitable change of variables G.

In order to study this Sbox, the authors introduced the law $\diamond\colon \mathbb{F}_2^3 \times \mathbb{F}_2^3 \to \mathbb{F}_2^3$ that is defined by:

$$\forall x, y \in \mathbb{F}_2^3, \quad x \diamond y := \begin{pmatrix} x_0 + y_0 + x_1 y_2 + x_2 y_1 \\ x_1 + y_1 \\ x_2 + y_2 \end{pmatrix}.$$

We can computationally verify that this Sbox is APN, however it has a probability-1 $\diamond$-differential $\Delta^{\text{in}} \xrightarrow{S,\diamond} \Delta^{\text{out}}$, where $\Delta^{\text{in}} = \texttt{0x6}, \Delta^{\text{out}} = \texttt{0x4}$. This can equivalently be understood as probability-1 commutative property, or as a probability-1 differential of S^G for some G.

We found out by hand that the change of variables G, that is defined by:

$$\forall x \in \mathbb{F}_2^3, \quad G(x) := \begin{pmatrix} x_0 + x_1 x_2 \\ x_1 + x_2 \\ x_2 \end{pmatrix}, \quad G^{-1}(x) = \begin{pmatrix} x_0 + x_2 + x_1 x_2 \\ x_1 + x_2 \\ x_2 \end{pmatrix},$$

satisfies the equality $x \diamond y = T^{G^{-1}}_{G(y)}(x)$ for any x, y. Because of Lemma 4.15, any isomorphism between $(\mathbb{F}_2^3, \diamond)$ and $(\mathbb{F}_2^3, +)$ can be chosen instead of G. Nonetheless, we continue focusing on this arbitrary case. In particular, we consider A and B that we define by:

$$A := T^{G^{-1}}_{G(\Delta^{\text{in}})} = x \diamond \texttt{0x6} = \begin{pmatrix} x_0 + x_1 + x_2 \\ x_1 + 1 \\ x_2 + 1 \end{pmatrix}, \quad \text{and}$$

$$B := T^{G^{-1}}_{G(\Delta^{\text{out}})} = x \diamond \texttt{0x4} = \begin{pmatrix} x_0 + x_1 \\ x_1 \\ x_2 + 1 \end{pmatrix}.$$

By Theorem 4.14, it holds that $S \circ A = B \circ S$. This can be verified from the ANF of S that is given below, together with the one of S^G:

$$S(x) := \begin{pmatrix} x_0 x_1 + x_1 x_2 + x_2 \\ x_0 + x_1 x_2 + x_1 \\ x_0 x_1 + x_0 x_2 + x_0 + x_2 \end{pmatrix}, \quad S^G(x) := \begin{pmatrix} x_0 + x_2 \\ x_0 x_1 + x_1 x_2 + x_1 + x_2 \\ x_0 x_1 + x_0 + x_1 x_2 \end{pmatrix}.$$

We also easily observe that the differential $\texttt{0x5} \xrightarrow{S^G} \texttt{0x6}$ holds with probability 1. This is again due to Theorem 4.14, because $G(\Delta^{\text{in}}) = \texttt{0x5}$ and $G(\Delta^{\text{out}}) = \texttt{0x6}$.

The toy cipher of [CCI24a, CCI24b]. The recent work of Calderini, Civino & Invernizzi [CCI24a, CCI24b] deals with the resistance against $\diamond$-differential cryptanalysis of Sboxes which are optimal with respect to standard differential cryptanalysis. They in particular show that such Sboxes have no reason to be optimal for other laws $\diamond$, and among an affine equivalence class, two distinct Sboxes can have two distinct uniformities with respect to $\diamond$.

To illustrate their work, they build an SPN similar to the previous one, this time with a 16-bit block size that is decomposed into 4 cells of four bits. The used 4-bit Sbox S is given in Table 4.8.

x	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
$S(x)$	0	e	b	1	7	c	9	6	d	3	4	f	2	8	a	5

Table 4.8: The Sbox used in [CCI24a, CCI24b].

Its differential uniformity is $\delta_S = 4$, but, for a law $\diamond$ built in the same way as before, it has a probability-1 $\diamond$-differential $\Delta^{\text{in}} \xrightarrow{S,\diamond} \Delta^{\text{out}}$, where $\Delta^{\text{in}} = \texttt{0x7}, \Delta^{\text{out}} = \texttt{0x6}$. The law is defined[4] by:

$$\forall x, y \in \mathbb{F}_2^4, \quad x \diamond y := \begin{pmatrix} x_0 + y_0 + x_2 y_3 + x_3 y_2 \\ x_1 + y_1 \\ x_2 + y_2 \\ x_3 + y_3 \end{pmatrix}.$$

This corresponds to a commutation with probability 1 of $A = \mathcal{T}_{\texttt{0x7}}$ and $B = \mathcal{T}_{\texttt{0x6}}$ through S, or a probability-1 differential $G(\Delta^{\text{in}}) \xrightarrow{S^G} G(\Delta^{\text{in}})$ for a suitable G. We can for instance use:

$$G := \begin{pmatrix} x_0 + x_2 x_3 + x_3 \\ x_1 \\ x_2 \\ x_3 \end{pmatrix}.$$

[4] A look-up table can be found in the slides of the presentation of [CCI24b], see https://wcc2024.sites.dmi.unipg.it/SLIDES/Invernizzi.pdf.

4.5 Concluding remarks

As shown in this chapter, the kit of cryptanalysis techniques which is available to an adversary is very large, in particular when the balance of a design between cost, efficiency, and security is biased toward efficiency. Among the attacks against the recent lightweight designs, invariants attacks are some of the most devastating. They indeed enable an adversary not only to leverage the simplicity of the round function, but also the simplicity of the key schedule. In this way, conditions on the master key can immediately be derived from conditions on the round keys.

While an invariant is simply a Boolean function which is constant on the cycles of the considered cryptographic function F, its existence can never been ruled out. In the case of a block cipher $\mathcal{E} = (E_k \colon \mathbb{F}_2^n \to \mathbb{F}_2^n)$, the question is instead to find ways of determining not only an invariant for a given E_k, but one which is common to E_k for many different keys. Furthermore, such an invariant should be easily evaluated in order to distinguish the cipher from a random permutation. As shown in Section 4.2.3, the framework of Beyne [Bey18] or the one of Beierle, Canteaut, & Leander [BCL18] are two reliable ways of finding such invariants.

Study of conjugates, yet another endless work. In particular, the latter one opened the question of the study of the conjugacy class of a cipher. Indeed, an invariant of F is (in most of the cases) a linear combination of $(\mathbf{1}_\sigma)_{\sigma\in\mathcal{C}}$ where $\mathcal{C}$ is the cycle decomposition of F. However, the cycle type is invariant under conjugacy, so finding an invariant for F can as well be done by studying any of its conjugates F^G. The authors point out that the study of invariants is equivalent to the linear cryptanalysis of all conjugates F^G. However, the study of conjugates is not only limited to the linear case: after all, an adversary is free to choose the system of coordinates which is better suited to their approach. This is the reason why we mainly focused on applying to conjugate ciphers the other major class of attacks, that is, differential attacks. With the example of Midori, we proved that such considerations are as prolific as in the linear case. This is actually a frightening observation as the differential uniformity or linearity of the conjugates of an Sbox are never studied by designers. While the actual security against linear or differential cryptanalysis of an original cipher $\mathcal{E}$ is already hard to study, this opened door to all conjugates seems endless.

Not so worrying after all ? The partial good news is that a random change of variables G is unlikely to provide a stronger distinguisher than the one obtained from the sole analysis of $\mathcal{E}$. Indeed, the conjugate linear layer is *a priori* not linear anymore and is expected to play the role of a single and very wide Sbox. Furthermore, while the constant/key addition is supposed to be the easiest layer to handle in the case of linear or differential cryptanalysis, its conjugate T_c^G becomes in general impressively intricate. This sketched analysis seems to restrict analysis of conjugates to very structured change of variables, which are, for instance, parallel applications of a cell-size change of variables, and/or to G which are very sparse and/or of low degree. Furthermore, in the light of the previous sections, $\diamond$-differential [CBS19] is *equivalent* to the differential analysis of conjugate ciphers. In particular, they do not form two new kinds of attacks, but a single one. From our point of view, conjugacy remains a more appropriate method to study such phenomena, as the adjustment of standard methods seems more direct.

Expanding our current security notions. Still it seems necessary to get a better understanding of the resistance against such attacks, and in particular of the differential uniformity and linearity of conjugate Sboxes. Even if a smaller uniformity or linearity cannot be ruled out for conjugates, being aware of such a property could enable designers to select the linear layer so that it does not behave well with system of coordinates in which the Sbox is weak.

Key schedule against weak-key attacks. With another approach, having a denser key schedule can also prevent weak-key attacks. The downsize this time is that this component, which is generally not the most studied, now deserves a proper analysis. It is however likely that after a precise analysis of it, weak-key spaces can still be figured out. For example, it would not be that surprising that the recent alternative representation of the AES key schedule eventually leads to *ad hoc* attacks leveraging it.

Chapter 5

Commutative cryptanalysis and its application to Midori and Scream

In the previous chapter, the study of *commutation* relations of the form $F \circ A = B \circ F$ was shown to be a way of studying differential properties of conjugate functions F^G by only focusing on the original function F. In this context, the nature of the commutants A, B is very constrained as they necessary are involutions without fixed point. Still, the generalization of such a study to any kind of commutants is really tempting.

Such a generalization happens to fall under an even more global theoretical framework presented by Wagner [Wag04] 20 years ago. However, while unifying cryptanalysis techniques enables us to put things into perspective, it should be done in such a way that the obtained class can be populated with actual examples of cryptanalysis. For this reason, and inspired by the example of Midori developed in Chapter 4, we choose in this chapter to only consider commutative relations where the commutants are *affine bijective mappings.* As it is, commutative cryptanalysis already generalizes differential cryptanalysis in a direction orthogonal to higher-order differential cryptanalysis that was presented in Chapter 3. Indeed, such a study is instead focused on the inherent symmetries of a block cipher. This kind of properties has naturally attracted much attention in the past [Bou+10, LMR15, Cha+17]. In particular, linear self-similarities [Bou+10, LMR15] have highlighted many weaknesses of cryptographic constructions, even in contexts where the actual used Sbox can be ignored. By allowing ourselves to consider affine mappings, we necessarily explore a slightly more general class in which unknown weaknesses can be expected to arise. Furthermore, we also target distinguishers that *do depend* on the actual used Sbox.

In this chapter, we then start again with the already well-developed case of Midori, which serves as our first example of commutation relations with affine commutants. From there, commutative cryptanalysis is introduced and a precise review of the techniques that fall under this framework is drawn. Then, the commutation properties of each individual layer of a standard SPN are studied in detail. In particular, the deterministic case is precisely grasped, while the different degrees of freedom given by the probabilistic case are presented. Subsequently, applications are given, the first one being naturally our narrative arc, Midori64. But we also show that commutative cryptanalysis enables us to discover similar deterministic distinguishers for Midori128, but also for the block cipher Scream. We also present experimental results regarding probabilistic commutation which are less conclusive, but which indisputably point out an interesting direction. Finally, we take a step back and again question the nature of the obtained distinguishers by comparing them to existing frameworks, and in particular to the differential cryptanalysis. Finally, the chapter is concluded with some perspectives.

This chapter is based on a joint work with Patrick Felke, Gregor Leander, Patrick Neumann, Léo Perrin & Lukas Stennes that is published at IACR Transactions on Symmetric Cryptology, 2022(4) [Bau+23], and on a on-going work with same coauthors, together with Christof Beierle.

Contents

5.1 A broad yet applicable framework

5.1.1 Generalizing our previous example

In the previous chapter, the idea of studying commutation relations of the form $B \circ F = F \circ A$ through a cryptographic function $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ emerged from the analysis of conjugate ciphers. From now on, we only consider a block cipher $\mathcal{E} = (E_k\colon \mathbb{F}_2^n \xrightarrow{\sim} \mathbb{F}_2^n)$, and not its conjugates anymore. The objective of this section is to develop the analysis of the commutative properties of a block cipher by, again, relying on its iterated structure. For now on, we focus on the deterministic case; the probabilistic one is addressed in Section 5.3. Our goal is to detect the existence of two functions $A, B\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ such that for many keys $k \in \mathbb{F}_2^\kappa$, it holds that:

$$B \circ E_k = E_k \circ A. \tag{5.1}$$

Rather than generalizing at all costs, we prefer limiting ourselves to a manageable setting from which actual attacks can be mounted. This setting is extrapolated from the only known example at this point.

Example 5.1 (The commutant of Vert)**.** In the previous section, we showed that the differential $\nabla \to \nabla$ holds with probability 1 through the conjugates of all round functions of Midori under weak-key assumptions. From Proposition 4.6, this implies that $\mathcal{A} := T_{\nabla}^{\mathcal{G}^{-1}}$ commutes with any $F_k^{(r)}$, that is:

$$\forall\, r \in [\![0, R-1]\!]\,, \quad \mathcal{A} \circ F_k^{(r)} = F_k^{(r)} \circ \mathcal{A}.$$

From this property for the individual rounds, the same behaviour is deduced for the whole cipher as:

$$\begin{aligned}
E_k \circ \mathcal{A} &= F_k^{(r)} \circ \ldots \circ F_k^{(1)} \circ F_k^{(0)} \circ \mathcal{A} \qquad (5.2)\\
&= F_k^{(r)} \circ \ldots \circ F_k^{(1)} \circ \left(F_k^{(0)} \circ \mathcal{A}\right)\\
&= F_k^{(r)} \circ \ldots \circ F_k^{(1)} \circ \left(\mathcal{A} \circ F_k^{(0)}\right)\\
&= F_k^{(r)} \circ \ldots \circ \left(F_k^{(1)} \circ \mathcal{A}\right) \circ F_k^{(0)}\\
&= F_k^{(r)} \circ \ldots \circ \left(\mathcal{A} \circ F_k^{(1)}\right) \circ F_k^{(0)}\\
&= \cdots\\
&= \mathcal{A} \circ E_k.
\end{aligned}$$

Furthermore, we already noticed that $\mathcal{A}$ is an involution (with no fixed point), and in particular a bijective mapping. We also observed in Eq. (4.30), that $\mathcal{A}$ is the parallel application of the affine mapping A whose ANF is recalled in Eq. (5.3).

$$A(x_0, x_1, x_2, x_3) = \begin{pmatrix} x_2 + 1 \\ x_1 + 1 \\ x_0 + 1 \\ x_3 + 1 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{pmatrix} + \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} \tag{5.3}$$

▷

Based on Example 5.1, we therefore only focus on commutants that are *affine bijective mappings*. Among this class, the ones that are *parallel applications of a single affine bijective mapping*, as for example $\mathcal{A}$, will play a crucial role. We show in Section 5.1.2 that this restricted setting is already broad-enough to encompass many existing cryptanalysis techniques. It is also small-enough to develop a theory from which concrete applications can be found in practice. Note that the deterministic case is *on paper* well-known. Indeed, we are in that case searching for instanciated ciphers E_k that are affine-equivalent to themselves. In the following, given an affine mapping $A\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$, we denote by L_A its *linear part* $L_A := A + A(0)$ and by $c_A \in \mathbb{F}_2^n$ its *constant term*, *i.e.* $c_A := A(0)$.

Finally, we widen the scope of Eq. (5.2), by not only considering a single commutant $\mathcal{A}$. Instead, we allow a sequence of commutants $(\mathcal{A}^{(0)}, \dots, \mathcal{A}^{(R)})$ to satisfy, for many keys $k \in \mathbb{F}_2^\kappa$:

$$\forall\ r \in [\![0, R-1]\!]\,, \mathcal{A}^{(r+1)} \circ F_k^{(r)} = F_k^{(r)} \circ \mathcal{A}^{(r)}. \tag{5.4}$$

Such a property is a sufficient condition to find a pair of commutants for the whole cipher as it implies, in a similar manner to Eq. (5.2), that $\mathcal{A}^{(R)} \circ E_k = E_k \circ \mathcal{A}^{(0)}$. Such a sequence $(\mathcal{A}^{(0)}, \dots, \mathcal{A}^{(R)})$ is naturally called a *commutative trail* and Eq. (5.4) is denoted by:

$$\mathcal{A}^{(0)} \xrightarrow{F_k^{(0)}} \mathcal{A}^{(1)} \to \cdots \xrightarrow{F_k^{(R-1)}} \mathcal{A}^{(R)}. \tag{5.5}$$

5.1.2 A particular case of commutative diagram cryptanalysis

Before going further in the analysis of commutative cryptanalysis, it should be noted that the idea of studying equations of the form Eq. (5.1), in the deterministic or probabilistic case, is not new. The most prominent example is of course the case of differential cryptanalysis, in which case the commutants A, B are of the form $A = T_{\Delta^{\mathrm{in}}}$ and $B = T_{\Delta^{\mathrm{out}}}$ for some $\Delta^{\mathrm{in}}, \Delta^{\mathrm{out}} \in \mathbb{F}_2^n$. Constant addition being a particular case of bijective affine mapping, differential cryptanalysis is therefore a specific instantiation of commutative cryptanalysis.

Eq. (5.1) has also been studied much more generically 20 years ago by Wagner [Wag04]. The author indeed introduced a cryptanalysis technique based on *commutative diagrams* of the form:

$$\begin{tikzcd}
\mathbb{F}_2^n \arrow[r, "E_k"] \arrow[d, "\rho^{\mathrm{in}}"'] & \mathbb{F}_2^n \arrow[d, "\rho^{\mathrm{out}}"] \\
X \arrow[r, "\widetilde{E_k}"] & Y
\end{tikzcd},$$

where X, Y can be any set and $\widetilde{E_k}\colon X \to Y$ any suitable function. This framework is so general that it can embed almost any kind of attacks against block ciphers. For instance, linear cryptanalysis corresponds to the case where $X = Y = \mathbb{F}_2$,

$\rho^{\text{in}}, \rho^{\text{out}} \colon \mathbb{F}_2^n \to \mathbb{F}_2$ are linear Boolean functions and $\widetilde{E_k} = \text{Id}$. Commutative cryptanalysis, as we introduced it above, corresponds to the choice of $X = Y = \mathbb{F}_2^n$, with $\rho^{\text{in}} = A$, $\rho^{\text{out}} = B$ affine and bijective, and $\widetilde{E_k} = E_k$.

This implies that our framework is a *very specific case* of Wagner's one. We again stress that this is *on purpose*. Indeed, while *commutative diagram cryptanalysis* is a very elegant manner to approach a large class of cryptanalysis techniques, it has, to the best of our knowledge, never been instantiated using this methodology. The only examples that are known, *i.e.* almost all attacks against block ciphers, have been designed as very particular instances, with their very specific methodologies, and never by using the general framework. Stated otherwise, commutative diagram cryptanalysis was never used to discover any new interesting class of attacks. Furthermore, we are, as of today, far from giving security arguments against commutative diagram attacks in their general form.

In the light of this example, we show in the following section that commutation with affine bijective mappings is really close to some already known attacks, as it slightly generalizes them. This is the main reason why it can be effectively handled in Section 5.3, and populated with new examples in Section 5.4.

5.2 Related work

5.2.1 Differential cryptanalysis and some variants

As already explained above, a classical differential attack corresponds to the case of commutants of the form $A^{(r)} := T_{\Delta^{(r)}} \colon x \mapsto x + \Delta^{(r)}$, for any r. In that case, a commutative trail exactly corresponds to a classical differential trail. Some other generalizations of differential attacks fall also under the framework of commutative cryptanalysis.

Rotational(-XOR). Let ρ be the (cyclic) rotation of a word by one bit to the left. The concept of *rotational distinguisher* [KN10] consists in finding a pair of rotations ρ^i and ρ^j such that $\rho^i \circ F = F \circ \rho^j$ to distinguish a cryptographic function from a random one. Thus, this corresponds to the case where the commutants are rotations.

More recently, *rotational cryptanalysis* was generalized into *rotational-XOR (RX)* cryptanalysis [AL16]. In that case, the adversary is allowed to consider rotations that are possibly composed with constant additions. More precisely, the goal is to find (a, a', b, b') and ρ^i such that $F(x + a) + a' = F\left(\rho^i(x) + b\right) + b'$ holds with a high probability. Equivalently, if $\Delta^{\text{in}} := \rho^i(a) + b$ and $\Delta^{\text{out}} := a' + b'$, this means that the following equality should hold for many y:

$$F(y) + F\left(\rho^i(y) + \Delta^{\text{in}}\right) = \Delta^{\text{out}},$$

where $y \leftarrow x + a$ is used as change of variables. This is then a particular case of commutative property $B \circ F \approx F \circ A$, where $B = T_{\Delta^{\text{out}}}$, and $A = T_{\Delta^{\text{in}}} \circ \rho^i$.

In both attacks, such patterns are iteratively built; this again corresponds to a commutative trail.

c-differential. The concept of *c-differential* is introduced in [Ell+20]. It generalizes differentials in the following way.

Definition 5.2 (c-derivative [Ell+20])**.** Let p be a prime number and n, m positive integers. Let $\mathbb{F}_{p^n}$ (resp. $\mathbb{F}_{p^m}$) be the field with p^n (resp. p^m) elements. Given a p-ary (n, m)-function $F\colon \mathbb{F}_{p^n} \to \mathbb{F}_{p^m}$, and $c \in \mathbb{F}_{p^m}$, the (multiplicative) c-derivative of F with respect to $a \in \mathbb{F}_{p^n}$ is the function ${}_cD_aF$ defined by ${}_cD_aF(x) = F(x+a) - cF(x)$, for all $x \in \mathbb{F}_{p^n}$.
For a fixed $c \in \mathbb{F}_{p^m}$ let d be the maximal number of solutions of ${}_cD_aF(x) = b$, where the maximum is taken over $b \in \mathbb{F}_{p^m}, a \in \mathbb{F}_{p^n}$ ($a \in \mathbb{F}^*_{p^n}$, if $c = 1$). Then d is called the *c-differential uniformity* of F. ▷

When $p = 2$, the definition of ${}_cD_aF$ can thus be reformulated as:

$${}_cD_aF = F \circ T_a + M_c \circ F,$$

where $M_c := x \mapsto cx$. Stated otherwise, the c-derivative with respect to a estimates how much T_a and M_c commute through F. While c-differential uniformity has been extensively studied on its own, e.g. in [Ell+20, Ell+21, Mes+21, HPS22, Stă+22], we are not aware of any cryptanalysis leveraging it at this stage.

5.2.2 Self-similarity, linear commutants & invariants

The case of linear maps commuting with the round function of a cipher, or of a hash function, has already been addressed many times [Bou+10, LMR15, Cha+17]. It corresponds to the case where for any r, the commutant $A^{(r)}$ is linear rather than affine. As explained in [LMR15, Lemma 2], the existence of a linear mapping A that commutes with all unkeyed round functions sometimes leads to a *self-similarity*.

Definition 5.3 (Self-similarity [BB02, Bou+10])**.** The block cipher $\mathcal{E} = (E_k\colon \mathbb{F}_2^n \xrightarrow{\sim} \mathbb{F}_2^n)_{k \in \mathbb{F}_2^\kappa}$ admits a *self-similarity* relation if there exist efficiently computable bijections $G^{\text{in}}, G^{\text{out}}\colon \mathbb{F}_2^n \xrightarrow{\sim} \mathbb{F}_2^n$, and $G^{\text{key}}\colon \mathbb{F}_2^\kappa \xrightarrow{\sim} \mathbb{F}_2^\kappa$ such that:

$$\forall\ k \in \mathbb{F}_2^\kappa, \quad G^{\text{out}} \circ E_k = E_{G^{\text{key}}(k)} \circ G^{\text{in}}.$$

▷

As already mentioned in [Bou+10], a self-similarity property always leads to multiple interpretations. This can be used:

- either as a *related-key* distinguisher with probability 1 which holds for any key,
- or as a weak-key distinguisher with probability 1 which holds for *self-similar* keys, *i.e.* a key k that satisfies $k = G^{\text{key}}(k)$,
- but also as a complementation property, as the one of DES, which reduces the effective key size by 1 bit.

In the weak-key setting, this implies that for any weak key k, E_k satisfies $G^{\text{out}} = E_k \circ G^{\text{in}} \circ E_k^{-1}$, or, stated otherwise, that G^{in} and G^{out} are conjugate. Because of Proposition 4.4, they have as many fixed points. We denote by $\text{Fix}(F) := \{x \in \mathbb{F}_2^n, F(x) = x\}$ the set of fixed points of any mapping $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$. As a direct corollary of the following lemma, a self-similarity for which $G^{\text{in}} = G^{\text{out}}$ always implies the existence of a (possibly-empty) invariant spaces.

Lemma 5.4 (Iterated commutants and fixed points)**.** *Let $F, G, H\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$, with F bijective. Let us suppose that $F \circ G = H \circ F$. Let $i \in \mathbb{N}$. Then:*

$$F \circ G^i = H^i \circ F \quad \text{and} \quad F(\text{Fix}(G^i)) = \text{Fix}(H^i).$$

Proof. The first result comes from an immediate induction reasoning. Indeed, if $F \circ G^i = H^i \circ F$ holds, this implies that:

$$F \circ G^{i+1} = F \circ G \circ G^i = H \circ F \circ G^i = H \circ H^i \circ F = H^{i+1} \circ F.$$

Secondly, let $x \in \text{Fix}(G)$. Then $F(x) = F \circ G(x) = H \circ F(x)$. In other words, $F(x)$ is a fixed point of H and $F(\text{Fix}(G)) \subseteq \text{Fix}(H)$. The equality for $i = 1$ follows from the fact that $|\text{Fix}(G)| = |\text{Fix}(H)|$, since G and H are conjugates, while the general result is obtained with the same reasoning on the commutants G^i, H^i. □

We come back more deeply to the relationships between commutative cryptanalysis, self-similarities and invariants in Section 5.5.1.

In Definition 5.3, $G^{\text{in}}, G^{\text{out}}, G^{\text{key}}$ are *arbitrary* bijections, but in practical situations, they are linear (and now affine). Among the exceptions to this observation is the self-similarity of XTEA leveraged by Bouillaguet, Dunkelman, Leurent & Fouque [Bou+10]. In the following, we focus on some linear commutants and self-similarities exhibited in the previous works.

Cryptanalysis of Robin, iSCREAM and Zorro. In the work by Leander, Minaud & Rønjom [LMR15], and in more depth in the thesis by Minaud [Min16], the considered linear maps correspond to the so-called "Sbox independent setting". Indeed, they act as a permutation of the cells[1] and therefore commute with any Sbox layer. Such linear commutants therefore unveil how much symmetries in the overall construction of a primitive can be dangerous. Most of the known linear commutants are in fact related to symmetries of the linear layer and therefore independent of the actual Sbox. Among the exceptions to this observation is the self-similarity of $\mathcal{PURE}$ depicted in [Bou+10] which takes into account the fact that the Sbox is a power mapping.

The approach we take in Section 5.3 is rather complementary: our affine commutants are derived from a commutant A that satisfies $A \circ S = S \circ A$ and which is then Sbox-dependent. On the other hand, handling the linear layer is done at almost no cost by considering the parallel application of A on all, or almost all, cells.

[1] The definition in [LMR15] is actually more general than this one, to cope with partial Sbox layers where the Sbox is only applied to *some* of the cells. Yet, we do not consider this case in this thesis.

Cryptanalysis of NORX v2.0. While we are mainly focused on distinguishing properties, it should be noted that linear commutants have also been used in much more sophisticated attacks. The cryptanalysis of NORX v2.0 by Chaigneau, Fuhr, Gilbert, Jean & Reinhard [Cha+17] is among the most prominent examples. It consists in a ciphertext-only forgery attack on full NORX v2.0 whose cornerstone is a linear commutant. Indeed, the permutation F, that is applied to a square state, commutes with the rotation ρ of the state by one column to the left. The fact that $\rho \circ F = F \circ \rho$ is obtained by following a trail through two half-rounds as ρ commutes with the subcomponents $G_{\text{col}}, G_{\text{diag}}$ of F. It then holds that $\rho^i \circ F = F \circ \rho^i$ for any i, because of Lemma 5.4.

5.2.3 Differential backdoor over a two-round toy cipher

More recently, Beierle, Felke, Leander, Neumann & Stennes [Bei+23] presented a cipher that, for some weak keys, exhibits a probability-one differential over two rounds. This cipher corresponds to a differential version of the backdoored cipher Boomslang which was proposed in [Bei+22]. The Sbox layer $\mathcal{S}$ consists of the 3-time parallel application of a 5-bit Sbox S. The probability-one differential over two rounds was voluntarily hidden by making sure that it does *not* consist of a *single* differential trail.

In fact, it can be understood as a probability-one *commutative trail*. More precisely, there exist an affine map A and a difference Δ for which it holds that :

$$S \circ T_\Delta = A \circ S \quad \text{and } S \circ A = T_\Delta \circ S.$$

The full Sbox layer $\mathcal{S}$ therefore admits the same kind of property:

$$\mathcal{S} \circ T_\nabla = \mathcal{A} \circ \mathcal{S} \quad \text{and } \mathcal{S} \circ \mathcal{A} = T_\nabla \circ \mathcal{S},$$

where $\nabla = (\Delta, \Delta, \Delta)$ and $\mathcal{A} = A \times A \times A$.

Furthermore, $\mathcal{A}$ commutes with the linear layer $\mathcal{L} \colon \mathbb{F}_2^{15} \to \mathbb{F}_2^{15}$, i.e., $\mathcal{A} \circ \mathcal{L} = \mathcal{L} \circ \mathcal{A}$. Finally, with an arbitrary key $k^{(0)}$ and a weak key $k^{(1)}$, the aforementioned properties are combined to obtain an iterative probability-one commutative trail over two rounds, that is, over: $\mathcal{L} \circ \mathcal{S} \circ T_{k^{(0)}} \circ \mathcal{L} \circ \mathcal{S} \circ T_{k^{(0)}}$:

$$T_\nabla \xrightarrow{T_{k^{(0)}}} T_\nabla \xrightarrow{\mathcal{S}} \mathcal{A} \xrightarrow{\mathcal{L}} \mathcal{A} \xrightarrow{T_{k^{(1)}}} \mathcal{A} \xrightarrow{\mathcal{S}} T_\nabla \xrightarrow{\mathcal{L}} T_\nabla.$$

The notion of weak keys for commutative cryptanalysis is addressed in Section 5.3.1.a.

5.3 Commutation with a round function

The easiest way to find commutants for any iterative construction is to find compatible ones for each building block, and then chain those to form a trail, as depicted in Eqs. (5.2) and (5.5). We thus investigate each layer of a traditional SPN block cipher separately.

5.3.1 Commutation with a constant addition

5.3.1.a The general case

In the following, $c \in \mathbb{F}_2^n$ is a constant, and $A, B \colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ are two affine bijections. The round key (or round constant) addition has a non-trivial interaction with commutation. We indeed need to distinguish two quantities:

1. for a fixed triplet (c, A, B), the probability of $A \xrightarrow{T_c} B$, or equivalently the cardinality of $Z_{T_c}^{\mathrm{comm}}(A, B)$, and;
2. for a fixed pair (A, B), the number of constants $c \in \mathbb{F}_2^n$, for which $\left|Z_{T_c}^{\mathrm{comm}}(A, B)\right|$ is non-zero (or greater or equal to some value).

The first quantity indicates whether commutation holds or not in a fixed-key setting, while the second one measures the size of a set of weak keys. Both are addressed in the following proposition and its corollary.

Proposition 5.5. *Let $c \in \mathbb{F}_2^n$ and $A, B \colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ be affine bijections. Then:*

$$|Z_{T_c}^{\mathrm{comm}}(A, B)| = \begin{cases} 0 & \text{if } c + L_B(c) \notin \mathrm{Im}(A + B), \\ 2^{n-\mathrm{rk}(L_A+L_B)} & \text{otherwise.} \end{cases}$$

Equivalently, $A \xrightarrow{T_c} B$ holds with either probability 0 or $2^{-\mathrm{rk}(L_A+L_B)}$.

Proof. We consider the number of solutions of the following equation:

$$B \circ T_c(x) = T_c \circ A(x) \quad \Longleftrightarrow \quad (A + B)(x) = c + L_B(c). \tag{5.6}$$

This is an affine system of equations. It is then necessary and sufficient that $c + L_B(c) \in \mathrm{Im}(A + B)$ for this system to have solutions. When $c + L_B(c) \in \mathrm{Im}(A + B)$, the number of solutions is equal to the cardinality of $\ker(L_A + L_B)$, namely $2^{n-\mathrm{rk}(L_A+L_B)}$. □

In the light of Proposition 5.5, we introduce the following definition.

Definition 5.6 (Strong and p-weak keys)**.** Let A, B be affine bijections of $\mathbb{F}_2^n$. Let $c \in \mathbb{F}_2^n$. We say that c is *p-weak* (with respect to A, B) if $A \xrightarrow{T_c} B$ holds with probability p or equivalenlty if $\left|Z_{T_c}^{\mathrm{comm}}(A, B)\right| = p \cdot 2^n$. If $p = 0$, we simply say that c is *strong*. ▷

The higher the value of p is, the weaker is the considered key or constant. As a direct corollary of Proposition 5.5, the number of strong and weak keys can be computed.

Corollary 5.7. *Let $A, B\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ be affine bijections. Then, with respect to (A, B), there only exist strong or $2^{-\mathrm{rk}(L_A+L_B)}$-weak keys. Furthermore the number of $2^{-\mathrm{rk}(L_A+L_B)}$-weak keys is given by:*

$$|\mathrm{Im}(\mathrm{Id}+L_B) \cap \mathrm{Im}(A+B)| \times |\ker(\mathrm{Id}+L_B)| \ .$$

Proof. Because of Eq. (5.6), the set of weak keys W can be expressed as:

$$\begin{aligned} W &= \{c \in \mathbb{F}_2^n, c + L_B(c) \in \mathrm{Im}(A+B)\} \\ &= \bigsqcup_{y \in \mathrm{Im}(\mathrm{Id}+L_B)\cap\mathrm{Im}(A+B)} \{c \in \mathbb{F}_2^n, (\mathrm{Id}+L_B)(c) = y\} \\ &= \bigsqcup_{y \in \mathrm{Im}(\mathrm{Id}+L_B)\cap\mathrm{Im}(A+B)} (\mathrm{Id}+L_B)^{-1}(\{y\}) \end{aligned}$$

But for any $y \in \mathrm{Im}(\mathrm{Id}+L_B) \cap \mathrm{Im}(A+B)$ the preimage $(\mathrm{Id}+L_B)^{-1}(\{y\})$ has the same cardinality as $(\mathrm{Id}+L_B)^{-1}(\{0\}) = \ker(\mathrm{Id}+L_B)$. □

5.3.1.b The case where $L_A = L_B$

As we will see, the case where $L_A = L_B$ is of particular interest. We therefore adapt Corollary 5.7 to this specific case.

Corollary 5.8. *Let $A, B\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ be affine bijections such that $L_A = L_B$. Then a constant $c \in \mathbb{F}_2^n$ is 1-weak if $(\mathrm{Id}+L_B)(c) = c_A + c_B$, and strong otherwise.*

The first example of such a situation is given by a differential through constant addition.

Example 5.9 (The differential case)**.** Let us suppose that $A = T_{\Delta^{\mathrm{in}}}$ and $B = T_{\Delta^{\mathrm{out}}}$. In that case, $L_A = L_B = \mathrm{Id}$ and any key is either strong or 1-weak, *i.e.* the weakest possible from a security standpoint. Furthermore, as $\mathrm{Id} + L_B = 0$, if $c_A + c_B = 0$, *i.e.* if $\Delta^{\mathrm{in}} = \Delta^{\mathrm{out}}$, then all keys are 1-weak. Otherwise, all keys are strong. This is consistent with the usual interpretation: for any $c \in \mathbb{F}_2^n$, the differential probability $\Delta^{\mathrm{in}} \xrightarrow{T_c} \Delta^{\mathrm{out}}$ holds with probability 1 if $\Delta^{\mathrm{in}} = \Delta^{\mathrm{out}}$ and 0 otherwise. Weakness and strength for commutation therefore generalize the differential case. ▷

However, Corollary 5.8 actually masks part of the actual situation. Indeed, we easily observe by hand that for any $c, x \in \mathbb{F}_2^n$ it holds that:

$$T_c \circ A(x) = L_A(x) + c + c_A = A\left(x + L_A^{-1}(c)\right) = A \circ T_{L_A^{-1}(c)}(x). \tag{5.7}$$

So we immediately deduce that the set of 1-weak keys for $A \xrightarrow{T_k} A$ is precisely the set $\left\{k \in \mathbb{F}_2^n, k = L_A^{-1}(k)\right\} = \mathrm{Fix}(L_A^{-1}) = \mathrm{Fix}(L_A)$. In the more general case where

$L_B = L_A$, we can write A as $A = B \circ T_{L_A^{-1}(c_A + c_B)}$ and we therefore obtain for any $c \in \mathbb{F}_2^n$:

$$T_c \circ A = A \circ T_{L_A^{-1}(c)} = B \circ T_{L_A^{-1}(c_A + c_B)} \circ T_{L_A^{-1}(c)} = B \circ T_{L_A^{-1}(c_A + c_B + c)}.$$

Again, we immediately deduce that the set of 1-weak keys for $A \xrightarrow{T_k} B$ with $L_A = L_B$ is the set $\left\{k \in \mathbb{F}_2^n, k = L_A^{-1}(c_A + c_B + k)\right\} = (\mathrm{Id} + L_A)^{-1}(\{c_A + c_B\})$.

However, a deeper look at Eq. (5.7) shows that this equation actually points out a related-key property: given a pair of related keys $(c, L_A(c))$, the encryption of any related pair of plaintexts $(x, A(x))$ is again, a related pair $(T_c(x), A(T_c(x)))$. This is detailed in the following proposition.

Proposition 5.10 (Related-key distinguisher with probability one)**.** *Let $\mathcal{E} = (E_k \colon \mathbb{F}_2^n \xrightarrow{\sim} \mathbb{F}_2^n)_{k \in (\mathbb{F}_2^n)^\kappa}$ be a key-alternating round cipher whose round functions are defined by $F_k^{(r)} := T_{k^{(r)}} \circ G^{(r)}$, where $G^{(r)} \colon \mathbb{F}_2^n \xrightarrow{\sim} \mathbb{F}_2^n$ and $r \in [\![0, R-1]\!]$. Let $\mathrm{KS} \colon (\mathbb{F}_2^n)^\kappa \to (\mathbb{F}_2^n)^R$ be the associated key schedule. Let $A \colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ be an affine bijection and $G^{\mathrm{key}} \colon (\mathbb{F}_2^n)^\kappa \to (\mathbb{F}_2^n)^\kappa$ a bijection such that:*

- $\forall r \in [\![0, R-1]\!]\,, \quad A \circ G^{(r)} = G^{(r)} \circ A$, *and*
- $\forall k \in (\mathbb{F}_2^n)^\kappa, \quad \mathrm{KS}(G^{\mathrm{key}}(k)) = \left(L_A(k^{(0)}), \ldots, L_A(k^{(R-1)})\right).$

Then it holds that:

$$\forall k \in \mathbb{F}_2^\kappa, \quad A \circ E_k = E_{G^{\mathrm{key}}(k)} \circ A.$$

As a particular case, if $k \in \mathbb{F}_2^\kappa$ satisfies $k = G^{\mathrm{key}}(k)$, then:

$$A \circ E_k = E_k \circ A.$$

Proof. Let $x \in \mathbb{F}_2^n$, and $k \in \mathbb{F}_2^\kappa$. It is sufficient to prove that for any $r \in [\![0, R-1]\!]$, it holds that:

$$A \circ F_k^{(r)}(x) = F_{G^{\mathrm{key}}(k)}^{(r)} \circ A(x).$$

Indeed, the output of round r being the input of round $r+1$, the general result is deduced by an immediate inductive reasoning. Let then $r \in [\![0, R-1]\!]$. We observe that, for any $x \in \mathbb{F}_2^n$, it holds that:

$$\begin{aligned}
F_{G^{\mathrm{key}}(k)}^{(r)} \circ A(x) &= T_{L_A\left(k^{(r)}\right)} \circ G^{(r)} \circ A(x) \\
&= T_{L_A\left(k^{(r)}\right)} \circ A \circ G^{(r)}(x) \\
&= L_A\left(G^{(r)}(x)\right) + L_A\left(k^{(r)}\right) + c_A \\
&= L_A\left(G^{(r)}(x) + k^{(r)}\right) + c_A \\
&= A\left(G^{(r)}(x) + k^{(r)}\right) \\
&= A \circ F_k^{(r)}(x).
\end{aligned}$$

□

This interpretation is not present in our published paper [Bau+23]. It was later drawn to our attention by Bonnetain [Bon23] to whom the author of this thesis is grateful. With the benefit of hindsight, this result is only a slight generalization of [LMR15, Lemma 2].

Remark 5.11. The actual definition of such a function G^{key} depends on the key schedule algorithm. However, in practice, G^{key} is heavily based on L_A. An example where G^{key} is the parallel application of L_A is given in Section 5.4.1.a. ▷

As in the linear case that is handled in [LMR15], Proposition 5.10 implies that a commutative trail $A \xrightarrow{F_k^{(0)}} A \to \cdots \xrightarrow{F_k^{(R-1)}} A$ can produce related-key or weak-key distinguishers with probability one. In order to satisfy the first assumption of Proposition 5.10, we continue by analyzing the unkeyed layers.

5.3.2 Commutation with an Sbox layer

5.3.2.a The case of a single Sbox

When looking at the Sbox level, finding (all) affine permutations A, B that satisfy $B \circ S = S \circ A$ is a particular case of the well-known problem of affine equivalence. Indeed, this can equivalently be rewritten as $B \circ S \circ A^{-1} = S$ or $S = B^{-1} \circ S \circ A$, and both equations imply that the considered Sbox S is affine equivalent to itself.

Hence, a general algorithm that solves for any $S, S' \colon \mathbb{F}_2^m \xrightarrow{\sim} \mathbb{F}_2^m$ the problem of determining (if it exists) a pair (A, B) such that $A \circ S \circ B = S'$ can be adapted to our need. Among them, the algorithm of Dinur [Din18] works only if the algebraic degrees of S and S' are equal to $\deg_{\mathrm{a}}(S) = \deg_{\mathrm{a}}(S') = m - 1$. The algorithm of Biryukov, De Cannière, Braeken & Preneel [Bir+03] is less time efficient but has the advantage to work for any pair of permutations, regardless of the degree. Its principle can be easily described. The images of A and B are guessed step by step and then propagated, either by linearity (if $A(x)$ and $A(y)$ are known then it must hold that $A(x) + A(y) = L_A(x + y)$), or by using the relation $A \circ S \circ B = S'$. Any time that a contradiction appears, the algorithm backtracks and starts with a new guess, until the first correct pair, if it exists, is returned.

By rather letting the algorithm exhaustively list all pairs of affine permutations, we can effectively recover the list of *all* (A, B) that satisfy $A \circ S \circ B = S'$. This works in practice for Sboxes of small size, *i.e.* $m \leq 8$. An implementation by Perrin of this algorithm is available in `sboxU` [Bau+24b] as the `self_affine_equivalent_mappings` function.

While a random permutation (of sufficient size) is not expected to be (non-trivially) affine equivalent to itself [Hou06], the Sboxes used in practice are usually highly structured, either because they correspond to a simple Boolean circuit for an efficient implementation, or because they have a strong mathematical structure, for instance, because they are affine-equivalent to a monomial over a finite field. Moreover, all known APN permutations admit in their CCZ-class a bijection that is linearly equivalent to itself. It is even conjectured by Beierle, Brinkmann & Leander [BBL21, Conjecture 1] that this should always hold. For 4-bit Sboxes, we checked all equivalence classes and found that 137 out of all 302 classes are non-trivially affine self-equivalent.

5.3.2.b The case of a full Sbox layer

Let us now consider the whole Sbox layer $\mathcal{S}$. If $\mathcal{S}$ consists of Sboxes S with non-trivial linearity or differential uniformity then [RP20, Corollary 1] implies that the only affine permutations $\mathcal{A}, \mathcal{B}$ such that $\mathcal{S} \circ \mathcal{A} = \mathcal{B} \circ \mathcal{S}$ are necessarily of the form:

$$\mathcal{A} = \mathcal{P} \circ (A^{(0)} \times \ldots \times A^{(s-1)}), \quad \mathcal{B} = \mathcal{P}' \circ (B^{(0)} \times \ldots \times B^{(s-1)});$$

where $\mathcal{P}, \mathcal{P}'$ are permutations of cells, and $A^{(i)}, B^{(i)} \colon \mathbb{F}_2^m \to \mathbb{F}_2^m$ are affine bijections that satisfy $S \circ A^{(i)} = B^{(i)} \circ S$ for any i. In other words, finding $\mathcal{A}$ and $\mathcal{B}$ can be reduced to finding A and B such that $S \circ A = B \circ S$ and combining them accordingly. By considering the trivial case where $A^{(i)} = B^{(i)} = \mathrm{Id}$ for all i, this also includes all permutations of cells as obvious commutants of the Sbox layer.

5.3.2.c The probabilistic case

For now on, we only mentioned deterministic commutation through the Sbox, but we are also interested in cases where $B \circ S \approx S \circ A$. Finding such a pair (A, B) of affine bijections is already mentioned in the work by Biryukov, De Cannière, Braeken & Preneel [Bir+03, Section 4.3] as the *almost affine equivalence problem*. They suggest to tweak their algorithm in such a way that it allows the user a margin of error: as long as $B \circ S \circ A^{-1}$ and S match for almost all values, the algorithm should not backtrack. This way, the obtained pairs satisfy $B \circ S \approx S \circ A$. Again, an implementation by Perrin is available in `sboxU` [Bau+24b] as the `self_affine_equivalent_mappings_approx` function.

The probabilistic problem for the full layer $\mathcal{S}$ is even more intricate and we did not address it. In our experiments presented in Section 5.4, we only consider parallel applications of almost-commuting mappings $A^{(i)}, B^{(i)}$ of the size of the Sbox.

5.3.3 Commuting with a linear layer

5.3.3.a The general case

Finally, it remains to study the case of commutation with a linear layer. We are therefore interested in the number of solutions of the following equation, where $\mathcal{L} \colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ is linear and $A, B \colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ affine:

$$\mathcal{L} \circ A(x) = B \circ \mathcal{L}(x) \quad \iff \quad (\mathcal{L} \circ L_A + L_B \circ \mathcal{L})\,(x) = \mathcal{L}(c_A) + c_B. \tag{5.8}$$

This number of solutions, *i.e.* the cardinality of $Z_{\mathcal{L}}^{\mathrm{comm}}(A, B)$, is again the number of solutions of an affine system. It therefore holds that:

$$|Z_{\mathcal{L}}^{\mathrm{comm}}(A, B)| = \begin{cases} 0 & \text{if } \mathcal{L}(c_A) + c_B \notin \mathrm{Im}(\mathcal{L} \circ L_A + L_B \circ \mathcal{L}), \\ 2^{n-\mathrm{rk}(\mathcal{L} \circ L_A + L_B \circ \mathcal{L})} & \text{otherwise.} \end{cases}$$

Computing the probability of this commutation property therefore sums up to linear algebra and to the computation of the dimension of the kernel of $\mathcal{L} \circ L_A + L_B \circ \mathcal{L}$.

5.3.3.b The deterministic case

If we require the commutation to happen with probability one, then it must hold that $\mathrm{rk}(\mathcal{L} \circ L_A + L_B \circ \mathcal{L}) = 0$, *i.e.* $\mathrm{Im}(\mathcal{L} \circ L_A + L_B \circ \mathcal{L}) = \{0\}$, or stated otherwise that $\mathcal{L} \circ L_A = L_B \circ \mathcal{L}$. This also implies that $\mathcal{L}(c_A) + c_B \in \{0\}$ and therefore $\mathcal{L}(c_A) = c_B$. Reciprocally, if both conditions are satisfied, then for any $x \in \mathbb{F}_2^n$, we have:

$$\mathcal{L} \circ A(x) = \mathcal{L}(L_A(x) + c_A) = \mathcal{L}(L_A(x)) + \mathcal{L}(c_A) = L_B(\mathcal{L}(x)) + c_B = B \circ \mathcal{L}(x).$$

We deduce the following lemma.

Lemma 5.12 (Commutation with a linear layer)**.** *Let $\mathcal{L} \colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ be linear and $A, B \colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ affine functions. Then:*

$$B \circ \mathcal{L} = \mathcal{L} \circ A \qquad \Longleftrightarrow \qquad L_B \circ \mathcal{L} = \mathcal{L} \circ L_A \quad \textit{and} \quad \mathcal{L}(c_A) = c_B.$$

If additionally, A and B are parallel applications of affine mappings of the size of the Sbox, then L_A and L_B are block diagonal matrices. We can therefore express the condition $L_B \circ \mathcal{L} = \mathcal{L} \circ L_A$ as commutation conditions on the submatrices.

Lemma 5.13 (Block diagonal commutants)**.** *Let $n = m \times s$. Let $\mathcal{L}$ be a block matrices of size $s \times s$ whose blocks are of size $m \times m$, i.e. $\mathcal{L} := (L_{ij})_{i,j \in [\![0,s-1]\!]}$ with $\mathcal{L}_{ij} \in \mathbf{M}_m(\mathbb{F}_2)$. Let $\mathcal{A}, \mathcal{B}$ be two block diagonal matrices that are decomposed as follows:*

$$\mathcal{A} = \begin{pmatrix} A^{(0)} & 0 & 0 & 0 \\ 0 & A^{(1)} & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & A^{(s-1)} \end{pmatrix}, \quad \mathcal{B} = \begin{pmatrix} B^{(0)} & 0 & 0 & 0 \\ 0 & B^{(1)} & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & B^{(s-1)} \end{pmatrix};$$

with $A^{(i)}, B^{(i)} \in \mathbf{M}_m(\mathbb{F}_2)$ for any i. Then:

$$\mathcal{B} \circ \mathcal{L} = \mathcal{L} \circ \mathcal{A} \qquad \Longleftrightarrow \forall\ i, j \in [\![0, s-1]\!]\,,\ B^{(i)} L_{ij} = L_{ij} A^{(j)}. \tag{5.9}$$

Proof. Let us denote the blocks of $\mathcal{A}$ and $\mathcal{B}$ by $\mathcal{A} = (A_{ij})_{i,j \in [\![0,s-1]\!]}$ and $\mathcal{B} = (B_{ij})_{i,j \in [\![0,s-1]\!]}$. From the matrix multiplication formula, we immediately get that $\mathcal{B} \circ L = L \circ \mathcal{A}$ if and only if:

$$\forall i, j \in [\![0, s-1]\!]\,, \quad \sum_{k=0}^{s-1} B_{ik} L_{kj} = \sum_{k=0}^{s-1} L_{ik} A_{kj}.$$

But the only term which is non-zero in the first sum (resp. in the second one) is $B_{ii} L_{ij} = B^{(i)} L_{ij}$ (resp. $L_{ij} A_{jj} = L_{ij} A^{(j)}$). In other words, the equality $\mathcal{B} \circ L = L \circ \mathcal{A}$ holds if and only if for any i, j, $B^{(i)} L_{ij} = L_{ij} A^{(j)}$. □

Therefore, if $\mathcal{A}$ and $\mathcal{B}$ are parallel applications of affine mappings of the size of the Sbox, then by combining Lemmas 5.12 and 5.13, the commutation of $\mathcal{A}, \mathcal{B}$ with a linear layer $\mathcal{L}$ sums up to verifying commutation relations for each block of the size of the Sbox, and a single evaluation of $\mathcal{L}$. This can thus be handled very easily, for instance using the algorithms described in Section 5.3.2. We can also restrict our search space to affine mappings that commute with probability one with the Sbox layer. These mappings, that are described in Section 5.3.2.b, are very structured. This structure can be used to filter and effectively search for commutants through the linear layer. This is described in detail in [Bau+23, Section 5].

5.3.3.c Linear layers with blocks in $\{0, \mathrm{Id}\}$

The very particular case where the investigated linear layer $\mathcal{L}$ is only made of either 0 or Id blocks of size $m \times m$ is important in the following. Indeed, in that case, Eq. (5.9) can be rewritten as:

$$\mathcal{B} \circ \mathcal{L} = \mathcal{L} \circ \mathcal{A} \quad \iff \forall\, i, j,\ L_{ij} \neq 0,\ B^{(i)} = A^{(j)}.$$

In particular, this implies that such a linear layer $\mathcal{L}$ always satisfies $\mathcal{A} \circ \mathcal{L} = \mathcal{L} \circ \mathcal{A}$ when $\mathcal{A}$ is the parallel application of *any single* linear mapping of the size of the Sbox. In case where $\mathcal{A}$ is the parallel application of a single *affine* mapping, the commutation relation holds if and only if the condition $\mathcal{L}(c_{\mathcal{A}}) = c_{\mathcal{A}}$ is satisfied for the constant $c_{\mathcal{A}}$.

As simple as this example might seem, it is actually very enlightening, as a lot of linear layers from the literature are built in this way. This is the case of binary MixColumns layers as in Midori and many other ciphers, but also of the linear layers of LS designs [Gro+15a]. One should thus be careful when defining a cipher using a self-affine-equivalent Sbox that satisfies $A \circ S = S \circ A$ for some affine bijection $A \colon \mathbb{F}_2^m \to \mathbb{F}_2^m$, together with a binary linear layer. This is the case of Vert and Scream whose analyses are detailed in the following section.

5.4 Applications

The main tools to build a commutative trail by a layer-by-layer analysis are now settled. This section describes some instantiations of this framework. We first address the most direct ones, which are commutative trails which hold with probability 1. Then only, we describe probabilistic ones for which theory and experimentation seem to be less consistent.

5.4.1 Deterministic applications

5.4.1.a Midori64

We naturally start by looking at Midori64, and more precisely at the distinguisher presented in Section 4.3.3 that can be interpreted as a commutative trail with probability 1 in the light of Section 4.4 and Example 5.1.

The affine bijective mapping $A\colon \mathbb{F}_2^4 \to \mathbb{F}_2^4$ that is described in Eq. (5.3) commutes with the Sbox S of Midori. It is returned in a matter of seconds by the function `self_affine_equivalent_mappings` function of `sboxU` [Bau+24b] applied to S.

Let us denote by $\mathcal{A}\colon \mathbb{F}_2^{64} \to \mathbb{F}_2^{64}$ its 16-time parallel application. Because of Section 5.3.2.b, we get $\mathcal{A} \circ \mathcal{S} = \mathcal{S} \circ \mathcal{A}$.

Regarding the linear layers, it naturally holds that $\mathcal{A} \circ \mathcal{P} = \mathcal{P} \circ \mathcal{A}$ for any permutation of cells $\mathcal{P}$, and in particular for $\mathcal{P} \in \{\mathsf{SC}, \mathsf{SR}\}$. Regarding the MixColumns layer, we first notice using Eq. (4.1) that $M\colon (\mathbb{F}_2^4)^4 \to (\mathbb{F}_2^4)^4$ can be represented as a binary block matrix whose blocks are either $0 \in \mathbf{M}_4(\mathbb{F}_2)$ or $\mathrm{Id} \in \mathbf{M}_4(\mathbb{F}_2)$. We also easily observe that for any $c \in \mathbb{F}_2^4$, it holds that:

$$M(c,c,c,c) = (c+c+c, c+c+c, c+c+c, c+c+c) = (c,c,c,c).$$

Because of Section 5.3.3.c, this implies that M commutes with the 4-time parallel application of *any* affine mapping over $\mathbb{F}_2^4$, and in particular, we obtain:

$$M \circ (A \times A \times A \times A) = (A \times A \times A \times A) \circ M.$$

We naturally deduce that the 4-time parallel application of M commutes with the 16-time parallel application of A, or in other words that $\mathcal{A} \circ \mathsf{MC} = \mathsf{MC} \circ \mathcal{A}$.

All in all, $\mathcal{A}$ commutes with a round of Midori without constant (and key) addition that remains to be studied.

As discussed in Section 5.3.1.b, A commutes with probability 1 with a constant addition T_c with $c \in \mathbb{F}_2^4$ if and only if c is a fixed point of L_A. Thanks to Eq. (5.3), we observe that:

$$\mathrm{Id} + L_A = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}. \tag{5.10}$$

Its kernel is then of dimension $4 - \dim(\mathrm{Im}(\mathrm{Id}+L_A)) = 4-1 = 3$, and we observe that $(0,1,0,0)$, $(0,0,0,1)$ and $(1,0,1,0)$ are independent and all belong to $\ker(\mathrm{Id}+L_A)$. Therefore, $\ker(\mathrm{Id}+L_A) = \langle \mathtt{0x2}, \mathtt{0x5}, \mathtt{0x8} \rangle$. Each nibble of a constant $C \in \mathbb{F}_2^{64}$ must then belong to $\langle \mathtt{0x2}, \mathtt{0x5}, \mathtt{0x8} \rangle$ for C to be considered weak. This does not hold for the original Midori64 as $1 \notin \langle \mathtt{0x2}, \mathtt{0x5}, \mathtt{0x8} \rangle$. We nonetheless conclude that the mapping $\mathcal{A}$ commutes with the unkeyed round function of any member of Vert^c, with $c \in \langle \mathtt{0x2}, \mathtt{0x5}, \mathtt{0x8} \rangle$.

Finally, we observe that applying $G^{\mathrm{key}} := L_{\mathcal{A}} \times L_{\mathcal{A}}\colon (\mathbb{F}_2^{64})^2 \to (\mathbb{F}_2^{64})^2$ to a master key $k \in (\mathbb{F}_2^{64})^2$ sums up to applying $L_{\mathcal{A}}$ to each of the round keys. We are then precisely in the scenario of Proposition 5.10 and we can then conclude that for any block cipher $\mathcal{E} = (E_k \colon \mathbb{F}_2^n \xrightarrow{\sim} \mathbb{F}_2^n)$ among the families Vert^c with $c \in \langle \texttt{0x2}, \texttt{0x5}, \texttt{0x8} \rangle$, it holds that:

$$\forall\ k \in \mathbb{F}_2^{128}, \quad \mathcal{A} \circ E_k = E_{G^{\text{key}}(k)} \circ \mathcal{A}.$$

The weak-key distinguisher of Section 4.3.3 is not only rediscovered, it can also be reinterpreted as a related-key one which holds for any key, or as a complementation property which reduces the effective size of the key by 1 bit.

Other commutants for Midori64. We can also be interested in the existence of other such distinguishers. By testing the Sbox S with `sboxU` [Bau+24b], the function `self_affine_equivalent_mappings` returns not only the pair (A, A), but also the pairs (B, C) and (C, B) where $B, C \colon \mathbb{F}_2^4 \to \mathbb{F}_2^4$ are defined by:

$$B(x) = \begin{pmatrix} x_2 \\ x_1 + 1 \\ x_0 \\ x_0 + x_2 + x_3 + 1 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 \end{pmatrix} \begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \end{pmatrix}, \text{ and}$$

$$C(x) = \begin{pmatrix} x_0 + 1 \\ x_1 \\ x_2 + 1 \\ x_0 + x_2 + x_3 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{pmatrix} \begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{pmatrix} + \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \end{pmatrix}.$$

This implies that the following relations hold with probability 1:

$$B \xrightarrow{S} C \quad \text{and} \quad C \xrightarrow{S} B.$$

In the general case, such a pair of commutations is not expected, but always happens in the case of an involutive Sbox. Indeed, in that case:

$$S \circ B = C \circ S \quad \iff \quad B \circ S^{-1} = S^{-1} \circ C \quad \iff \quad B \circ S = S \circ C,$$

where the first equivalence holds for any bijective Sbox and the second one for any involutive Sbox.

Among the notable properties of these specific B and C, we observe that they commute one with the other and that they both are involutive:

$$B \circ C = C \circ B = A \quad \text{and} \quad B^2 = C^2 = \mathrm{Id}.$$

However we also note that $\mathrm{Fix}(L_A) = \mathrm{Fix}(L_B) = \mathrm{Fix}(L_C)$. This implies that the set of weak constants or weak keys corresponding to the *alternating* commutative trail $\mathcal{B} \to \mathcal{C} \to \mathcal{B} \to \mathcal{C} \to \cdots \to \mathcal{B}$ is exactly the same as the one of the commutative trail $\mathcal{A} \to \cdots \to \mathcal{A}$ that is described above.

5.4.1.b Midori128

Due to the strong similarities between Midori64 and Midori128, it is tempting to also apply the commutative framework to the version with a bigger state.

Let us recall from Section 4.1.3 that Midori128 use four different Sboxes SSb_i, $i \in [\![0, 3]\!]$. For any i, the Sbox SSb_i is applied four times in parallel on the i-th row of the state. But as depicted in Figure 4.4, each of them is a conjugate of the form SS^{L_i} where SS is the parallel application of two 4-bit Sboxes and $L_i \colon \mathbb{F}_2^8 \to \mathbb{F}_2^8$ is a bit permutation, *i.e.* the matrix of L_i is a binary permutation matrix. By construction, it is clear that SS commutes with the *linear* bijection $A \colon \mathbb{F}_2^8 \to \mathbb{F}_2^8$ that swaps the most significant half and the least significant one. Indeed SS is a parallel application of nibble-wise Sboxes and A is the only non-trivial permutation of two nibbles. What is most striking is that this property is also shared by each L_i. Indeed, let us denote by $\sigma_i \colon [\![0, 7]\!] \xrightarrow{\sim} [\![0, 7]\!]$ the bijections such that for any $x \in \mathbb{F}_2^8$:

$$L_i(x_0, \dots, x_7) = (x_{\sigma_i(0)}, \dots, x_{\sigma_i(7)}).$$

From Figure 4.4, we observe that $\sigma_0, \sigma_1, \sigma_2, \sigma_3$ are defined by:

$$\sigma_0 = (1\ 5)(3\ 7), \qquad \sigma_1 = (0\ 5\ 2\ 3)(1\ 6\ 7\ 4),$$
$$\sigma_2 = (0\ 2)(1\ 3\ 5\ 7)(4\ 6), \qquad \sigma_3 = (0\ 7\ 2\ 1)(3\ 6\ 5\ 4).$$

Similarly if we denote by ρ the bijection defined by $\rho = (0\ 4)(1\ 5)(2\ 6)(3\ 7)$, then for any x we have:

$$A(x_0, \dots, x_7) = (x_{\rho(0)}, \dots, x_{\rho(7)}).$$

We furthermore observe that $\rho \circ \sigma_i \circ \rho^{-1} = \sigma_i$ for any $i \in [\![0, 3]\!]$ and this implies that $A \circ L_i = L_i \circ A$, and therefore $A \circ \mathrm{SSb}_i = \mathrm{SSb}_i \circ A$. All in all, we conclude that the 16-time parallel application of A, that we denote by $\mathcal{A}$ commutes with the full Sbox layer.

As shown above, MC commutes with the 16-time parallel application of any affine mapping and in particular with $\mathcal{A}$. Any permutation of cells naturally commutes with $\mathcal{A}$. Finally, we observe that $\mathrm{Fix}(L_A) = \{(x, x), x \in \mathbb{F}_2^4\} \subset \mathbb{F}_2^8$. However, none of the bytes of the round constants of Midori128 lie in $\mathrm{Fix}(L_A)$, but this is *by construction* the case of the round constants of any member of $\mathrm{Grün}^{\mathrm{c}}$, that is defined in Section 4.1.3, as soon as $c \in \mathrm{Fix}(L_A)$. From the sixteen 4-bit conditions, we finally derive a set of $2^{128-4\times16} = 2^{64}$ weak keys for the weak-key distinguisher.

5.4.1.c Scream

Scream [Gro+15b] is a 128-bit-state and 128-bit-key tweakable block cipher in the LS-design family. Its 128-bit state can be viewed as an 8×16 matrix. The Sbox layer consists in applying a unique 8-bit Sbox in parallel to each column, while the linear layer consists in applying a unique 16-bit linear permutation (called Lbox) to each row. At each round, round constants are added to the first row of the state, the key is added to the state (and the tweak, that we consider to be equal to 0 here, is added on the first 4 rows). For further details, we refer to the document submitted to the CAESAR competition [Cae13].

By testing self-affine equivalence, we find out that the used Sbox S is indeed equivalent to itself and it satisfies $A \circ S = S \circ A$ for the 8-bit affine bijection $A \colon \mathbb{F}_2^8 \to \mathbb{F}_2^8$, that is defined by:

$$A := x \mapsto \begin{pmatrix} 1 & . & . & . & . & . & . & . \\ . & 1 & . & . & . & . & . & . \\ . & . & 1 & . & . & . & . & . \\ . & . & . & 1 & . & . & . & . \\ . & 1 & 1 & 1 & 1 & . & . & . \\ . & . & . & . & . & 1 & . & . \\ . & . & . & 1 & . & . & 1 & . \\ . & 1 & . & 1 & . & . & . & 1 \end{pmatrix} \cdot \begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{pmatrix} + \begin{pmatrix} 1 \\ . \\ . \\ . \\ . \\ 1 \\ . \\ . \end{pmatrix},$$

where a dot stands for 0 for an easier reading.

We further observe that $\mathrm{Fix}(L_{\mathcal{A}}) = \langle \texttt{0x01}, \texttt{0x10}, \texttt{0x20}, \texttt{0x40}, \texttt{0x80} \rangle$. As the round constants are added on the least significant row, this means that each of their bytes belongs to $\{\texttt{0x00}, \texttt{0x01}\}$. But this is a subset of $\mathrm{Fix}\,(L_{\mathcal{A}})$, so each of the round-constant additions commutes with the 8-time parallel application of A, that we denote again by $\mathcal{A}$.

It remains to study the linear layer $\mathcal{L}$ using Lemma 5.12. We look at $\mathbb{F}_2^{128}$ as $(\mathbb{F}_2^8)^{16}$ so that each copy of $\mathbb{F}_2^8$ corresponds to a cell on which the Sbox is applied. This way, $\mathcal{L}$ is decomposed as an 16×16 matrix whose blocks are of size 8×8. Because $\mathcal{L}$ consists in the parallel application of a linear bijection on the rows while the Sboxes are applied column-wise, each block of $\mathcal{L}$ is either the block 0 or Id. As observed in Section 5.3.3.c, such a linear layer commutes with the parallel application of any single linear function, and in particular with $L_{\mathcal{A}}$. Furthermore, $c_{\mathcal{A}} = \texttt{0b00100001}$, so by looking at $c_{\mathcal{A}}$ as a 8×16 matrix, it is composed of two all-1 rows and six all-0 rows. But one can easily verify that the all-1 vector, and obviously the all-0 one, are fixed points of the Lbox of Scream. Indeed the matrix of the Lbox is given in [Gro+15b, Figure 1] and we observe that all of its rows have an odd Hamming weight. Therefore, $\mathcal{A}$ commutes with any component of the unkeyed round function. Finally, because the key schedule sums up to the addition of the master key at each round, Proposition 5.10 can be applied in the same way as before with $G^{\mathrm{key}} = L_{\mathcal{A}}$.

Unlike our attacks against Vert and Grün, the obtained self-similarity, and the corresponding related-key and weak-key distinguishers hold for the original primitive, and not a modified instance. While this self-similarity is linear, it has, to the best of our knowledge never been mentioned elsewhere. In all likelihood, the main reason is that previous methods such as the one by Leander, Minaud & Rønjom [LMR15] focus on symmetries that do not depend on the Sbox, while the one just presented is dependent on the actual Sbox. In terms of weak-key distinguisher, the set of weak keys that is derived from the self-similarity is a strict subset of the one obtained in [TLS19]: their non-linear invariant works whenever each byte of the key belongs to $\{x \in \mathbb{F}_2^8, x_1 = x_2 = 0\}$, while, in the weak-key setting, our distinguisher holds when each byte belongs to $\{x \in \mathbb{F}_2^8, x_1 = x_2 = x_3 = 0\}$, and therefore for $2^{128-3\times 16} = 2^{80}$ weak keys.

5.4.2 Probabilistic applications

5.4.2.a Probabilistic trails through the linear layer

As shown by the previous examples and by Proposition 5.10, the weak-key space related to the obtained self-similarities is the set $\mathrm{Fix}(G^{\mathrm{key}})$. In our cases, this function G^{key} (and the number of its fixed points) is always closely-related to $\mathcal{A}$ and $\mathrm{Fix}(L_{\mathcal{A}})$. It is then very tempting to modify $\mathcal{A}$ in order to increase the number of its fixed points. In the case where $\mathcal{A}$ is the parallel application of $A\colon \mathbb{F}_2^m \to \mathbb{F}_2^m$, as $\mathrm{Fix}(L_{\mathcal{A}}) = \bigtimes_{i=0}^{s-1} \mathrm{Fix}(L_A)$, the most natural way is to replace some applications of A by applications of the identity mapping. With the standard vocabulary used for differential attacks, this amounts to decrease the number of *active* Sboxes and it has the effect of lowering the constraints on the key and thus of increasing the number of weak keys. Switching from $\mathcal{A}$ to such a *partial* affine bijection $\widetilde{\mathcal{A}}$ has no effect on commutation with the Sbox layer as $\widetilde{\mathcal{A}} \xrightarrow{\mathcal{S}} \widetilde{\mathcal{A}}$ still holds with probability 1 if A satisfies $A \xrightarrow{S} A$ with probability 1. However, such a change does not guarantee that $\widetilde{\mathcal{A}} \xrightarrow{\mathcal{L}} \widetilde{\mathcal{A}}$ should also hold with probability 1 through the linear layer $\mathcal{L}$. In particular, while the permutation of cells was never taken into account, it now requires a dedicated study.

We illustrate these first probabilistic commutative properties by focusing *only* on the family $\mathsf{Vert}_{\mathsf{SR}}$. This choice is motivated by the fact that ShuffleCells was by design chosen to have fewer symmetries than the ShiftRows of the AES, and is in practice harder to study.

Probabilistic commutation with M. Let $i \in [\![0, 15]\!]$ and let us denote its binary decomposition by $i = \sum_{\ell=0}^{3} i_\ell 2^\ell$, with $i_\ell \in \{0, 1\}$. Let $A\colon \mathbb{F}_2^m \to \mathbb{F}_2^m$ be an affine mapping. In the following, we denote by $A^0 := \mathrm{Id}$ and $A^1 = A$ and define the partial layer $\widetilde{\mathcal{A}}^{\times i}$ as:

$$\widetilde{\mathcal{A}}^{\times i} := A^{i_0} \times A^{i_1} \times A^{i_2} \times A^{i_3}. \tag{5.11}$$

We study the probabilistic transition $\widetilde{\mathcal{A}}^{\times i} \xrightarrow{M} \widetilde{\mathcal{A}}^{\times j}$ for any $i, j \in [\![0, 15]\!]$. To do so, we look at Eq. (5.8) which can in that case be expressed as:

$$\underbrace{\begin{pmatrix} . & \mathbf{1}_{j_0}(i_1)N & \mathbf{1}_{j_0}(i_2)N & \mathbf{1}_{j_0}(i_3)N \\ \mathbf{1}_{j_1}(i_0)N & . & \mathbf{1}_{j_1}(i_2)N & \mathbf{1}_{j_1}(i_3)N \\ \mathbf{1}_{j_2}(i_0)N & \mathbf{1}_{j_2}(i_1)N & . & \mathbf{1}_{j_2}(i_3)N \\ \mathbf{1}_{j_3}(i_0)N & \mathbf{1}_{j_2}(i_1)N & \mathbf{1}_{j_3}(i_3)N & . \end{pmatrix}}_{B(i,j)} x = c_A \left(M \begin{pmatrix} i_0 \\ i_1 \\ i_2 \\ i_3 \end{pmatrix} + \begin{pmatrix} j_0 \\ j_1 \\ j_2 \\ j_3 \end{pmatrix} \right) ,$$

where $N := L_A + \mathrm{Id}$. Again, if the right-hand side of the previous equation is not in the image of $B(i,j)$, then $\widetilde{\mathcal{A}}^{\times i} \xrightarrow{M} \widetilde{\mathcal{A}}^{\times j}$ holds with probability 0. Otherwise, the cardinality of $Z_M^{\mathrm{comm}}(\widetilde{\mathcal{A}}^{\times i}, \widetilde{\mathcal{A}}^{\times j})$ is the same as the cardinality of $\ker(B(i,j))$, which can be deduced from the one of its image that is given in Table 5.1.

j i	1	2	4	8	3	5	6	9	a	c	7	b	d	e	f
1	2	2	2	2	3	3	3	3	3	3	4	4	4	**2**	3
2	2	2	2	2	3	3	3	3	3	3	4	4	**2**	4	3
4	2	2	2	2	3	3	3	3	3	3	4	**2**	4	4	3
8	2	2	2	2	3	3	3	3	3	3	**2**	4	4	4	3
3	3	3	3	3	**2**	4	4	4	4	4	3	3	3	3	2
5	3	3	3	3	4	**2**	4	4	4	4	3	3	3	3	2
6	3	3	3	3	4	4	**2**	4	4	4	3	3	3	3	2
9	3	3	3	3	4	4	4	**2**	4	4	3	3	3	3	2
a	3	3	3	3	4	4	4	4	**2**	4	3	3	3	3	2
c	3	3	3	3	4	4	4	4	4	**2**	3	3	3	3	2
7	4	4	4	**2**	3	3	3	3	3	3	2	2	2	2	1
b	4	4	**2**	4	3	3	3	3	3	3	2	2	2	2	1
d	4	**2**	4	4	3	3	3	3	3	3	2	2	2	2	1
e	**2**	4	4	4	3	3	3	3	3	3	2	2	2	2	1
f	3	3	3	3	2	2	2	2	2	2	1	1	1	1	**0**

Table 5.1: Dimension of $\mathrm{Im}(B(i,j))$. All entries must be multiplied by $\dim(\mathrm{Im}(N))$. When written in bold, the commutation holds for all c_A, otherwise, we need $c_A \in \mathrm{Im}(N)$.

Commutation of a partial layer with SR. In order to choose appropriate values for i and j, we look at commutation through the ShiftRows operation. Following the cell numbering given in Figure 4.1, ShiftRows lets unchanged (among others) the cells of indices 0 and 8, and exchanges the values of the cells of indices 2 and 10.

As a consequence, if we consider $\widetilde{\mathcal{A}}$ that applies the identity mapping everywhere except on nibbles with index 0, 2, 8 and 10 where the same mapping $\mathcal{A}$ is applied, then $\widetilde{\mathcal{A}} \xrightarrow{\mathsf{SR}} \widetilde{\mathcal{A}}$ holds with probability 1. By generalizing the block matrix notation to affine mappings, the previously defined $\widetilde{A}$ can be described as:

$$\widetilde{A} = \begin{pmatrix} A & . & A & . \\ . & . & . & . \\ A & . & A & . \\ . & . & . & . \end{pmatrix}, \tag{5.12}$$

where a dot stands for the identity mapping $Id \colon \mathbb{F}_2^4 \to \mathbb{F}_2^4$. This "square" activity pattern is not new, and as for instance already been used in an attack against Prince [Can+15].

We can generalize such a commutation with probability 1 with the ShiftRows operation by considering two disjoint sets $Z_{\mathrm{Id}}, Z_A \subset [\![0, 15]\!]$ that partition $[\![0, 15]\!]$. If Id (resp. A) is applied to any cell with index $i \in Z_{\mathrm{Id}}$ (resp. $i \in Z_A$), and if Z_{Id} (resp. Z_A) is a union of supports of cycles of SR, then commutation between $\widetilde{A}$ and SR holds with probability 1.

However the pattern described in Eq. (5.12) has the advantage to apply the identity mapping on columns 1 and 3. This implies that the probability of $\widetilde{A} \xrightarrow{\mathsf{MC}} \widetilde{A}$ for such a mapping $\widetilde{A}$ only depends on the probability of the commutative trail $(A \times \mathrm{Id} \times A \times \mathrm{Id}) \xrightarrow{M} (A \times \mathrm{Id} \times A \times \mathrm{Id})$: it is actually the square of it.

Probabilistic commutative trail for Vert. In order to compute an actual probability of the trail over one round, we must choose a specific mapping A. We continue using the one described in Eq. (5.3) that satisfies $A \xrightarrow{\mathcal{S}} A$ with probability 1, and naturally implies that $\widetilde{A} \xrightarrow{\mathcal{S}} \widetilde{A}$ also holds with probability 1.

Then, in order to study the transition $(A \times \mathrm{Id} \times A \times \mathrm{Id}) \xrightarrow{M} (A \times \mathrm{Id} \times A \times \mathrm{Id})$, we examine the value at coordinate $(\mathtt{0x5}, \mathtt{0x5})$ of Table 5.1, because $\widetilde{A}^{\times 5} = A \times \mathrm{Id} \times A \times \mathrm{Id}$ according to the notation introduced in Eq. (5.11). As observed in Eq. (5.10), in this specific case $N = L_A + \mathrm{Id}$ has an image set of dimension 1, and thus $\mathrm{Im}(B(i,j))$ has dimension 2. Therefore, $\widetilde{A} \xrightarrow{\mathsf{MC}} \widetilde{A}$ occurs with probability $(2^{-2})^2 = 2^{-4}$ because of the two active columns, and $\widetilde{A} \xrightarrow{\mathsf{MC} \circ \mathsf{SR} \circ \mathcal{S}} \widetilde{A}$ holds with the same probability. Note that this probability holds without any heuristic argument. Indeed, for any $x \in \mathbb{F}_2^{64}$, $\widetilde{A} \circ \mathsf{MC} \circ \mathsf{SR} \circ \mathcal{S}(x) = \mathsf{MC} \circ \mathsf{SR} \circ \mathcal{S} \circ \widetilde{A}(x)$ if and only if $\widetilde{A} \circ \mathsf{MC}(y) = \mathsf{MC} \circ \widetilde{A}(y)$ where $y = \mathsf{SR} \circ \mathcal{S}(x)$.

Regarding key and constant additions, for any $C \in (\mathbb{F}_2^4)^{16}$, the commutation $\widetilde{A} \xrightarrow{T_C} \widetilde{A}$ holds with probability 1 if and only if the nibbles C_0, C_2, C_8, C_{10} all belong to $\mathrm{Fix}(L_A)$. In particular, for any $c \in \mathrm{Fix}(L_A)$, all round constants of $\mathsf{Vert}_{\mathsf{SR}}^c$ are weak. With four 1-bit conditions on each half of the master key k, each round key is also weak.

Thus, *assuming independence of the rounds*, we expect $\mathsf{Vert}_{\mathsf{SR}}^c$, with $c \in \mathrm{Fix}(L_A)$, to have a commutative trail $\widetilde{A} \xrightarrow{F_k^{(0)}} \widetilde{A} \to \cdots \xrightarrow{F_k^{(R-1)}} \widetilde{A}$ that holds with probability 2^{-4R}, where R is the number of full rounds, that is, the rounds including MC. This probabilistic behaviour is counter-balanced by the size of the space of 2^{-4R}-weak keys which is now significantly bigger with $2^{128-2\times 4} = 2^{120}$ keys.

Remark 5.14. The estimated probability is the one for the trail $\widetilde{\mathcal{A}} \xrightarrow{F_k^{(0)}} \widetilde{\mathcal{A}} \to \cdots \xrightarrow{F_k^{(R-1)}} \widetilde{\mathcal{A}}$. As in the differential case, it gives a lower bound on the probability of the commutation $\widetilde{\mathcal{A}} \xrightarrow{E_k} \widetilde{\mathcal{A}}$, which is in practice the only probability that can be estimated by an adversary with access given to plaintexts and ciphertexts. ▷

Experimental Results. These high probabilities enable us to experimentally test our distinguishers for round-reduced versions. The two main experiments are described below. The first one studies the probability of $\widetilde{\mathcal{A}} \xrightarrow{E_k} \widetilde{\mathcal{A}}$, but also of $\widetilde{\mathcal{A}} \xrightarrow{F_k^{(0)}} \widetilde{\mathcal{A}} \to \cdots \xrightarrow{F_k^{(R-1)}} \widetilde{\mathcal{A}}$ thanks to the full access given as experimenters. These experiments were made using the genuine key schedule of Midori.

Experiment 5.15 (Random weak key, random plaintext)**.** We pick uniformly at random some pairs (k, x) made of a weak key and a plaintext. For such a pair, we verify whether $\widetilde{\mathcal{A}} \circ E_k(x) \overset{?}{=} E_k \circ \widetilde{\mathcal{A}}(x)$ holds or not. This enables us to estimate the averaged probability of the commutation over all weak keys. We also verify whether or not the following equations are satisfied by (k, x):

$$\forall i \in \{0, \cdots, R-1\}, \quad F_k^{(i)} \circ \cdots \circ F_k^{(0)} \circ \widetilde{\mathcal{A}}(x) = \widetilde{\mathcal{A}} \circ F_k^{(i)} \circ \cdots \circ F_k^{(0)}(x).$$

This enables us to estimate the averaged probability of the trail over all weak keys. ▷

Example 5.15 was conducted for round-reduced versions of $\mathsf{Vert}^2_{\mathsf{SC}}$ and $\mathsf{Vert}^0_{\mathsf{SC}}$. Each time, the draw of a pair (x, k) was repeated 2^{36} times. We thus expected an average number of solutions for the trail $\widetilde{\mathcal{A}} \xrightarrow{F_k^{(0)}} \widetilde{\mathcal{A}} \to \cdots \xrightarrow{F_k^{(R-1)}} \widetilde{\mathcal{A}}$ of $2^{36-4(R-1)}$ for the R-round version, as the last round only consists of a single Sbox layer. Our results are depicted in Figure 5.1. As we can see on this figure, the behaviour

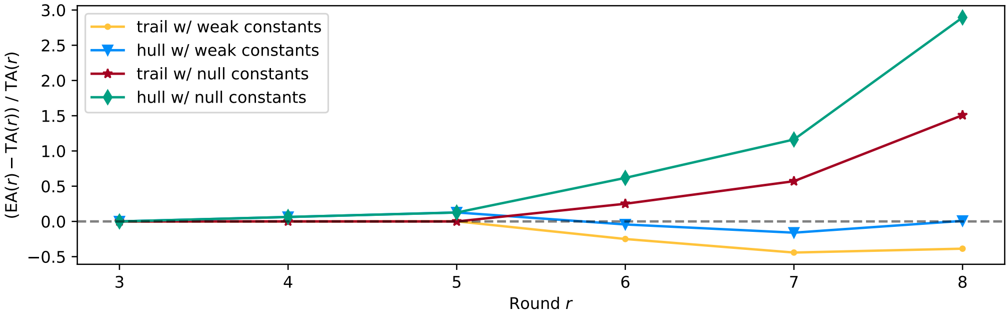


Figure 5.1: Evolution of the deviation between experimental (EA) and theoretical (TA) averages throughout the rounds.

of the experimental average is more intricate as one could first think. In the weak-constant setting and as the number of rounds increases, the experimental average probability of the trail seems to slowly decrease compared to the theoretical average. However, the experimental average over weak keys for the probability of $\widetilde{\mathcal{A}} \xrightarrow{E_k} \widetilde{\mathcal{A}}$ stays really close to the theoretical average for the trail. This seems to indicate the not-so-surprising fact that the round independence hypothesis is probably too strong. However, as a lower bound on the probability of $\widetilde{\mathcal{A}} \xrightarrow{E_k} \widetilde{\mathcal{A}}$, the probability of 2^{-4R} is satisfactory. Compared to the differential case, it seems that the "dominance" of the trail slowly vanishes, while the clustering effect becomes stronger and compensates this drop.

The distinctive case of the zero constants is also pointed out. Indeed, the zero constants form a particular class of weak constants as $0 \in \mathrm{Fix}(L_{\mathcal{A}})$. It corresponds to the case where no addition of round constant occurs at all. In that case, our results indicate that the probability of the trail $\widetilde{\mathcal{A}} \xrightarrow{F_k^{(0)}} \widetilde{\mathcal{A}} \to \cdots \xrightarrow{F_k^{(R-1)}} \widetilde{\mathcal{A}}$ is underestimated. It may be possible to explain this fact by looking at the cipher reduced to $2R$ *full* rounds, which in that case, can be expressed as the R-time composition of the function $F_k^{(0)} \circ F_k^{(1)}$. Such successive iterations of the *exact same round function* may be the reason why the probability differs from the average behavior.

We also studied scenarios where multiple plaintexts are encrypted using the same weak keys.

Experiment 5.16 (Fixed weak key, random plaintexts)**.** We pick uniformly at random a weak key k and a *set* of plaintexts $\mathbb{X}$ and count the number of $x \in \mathbb{X}$, for which:

$$\widetilde{\mathcal{A}} \circ E_k(x) = E_k(\widetilde{\mathcal{A}}(x))$$

is satisfied and the number of x for which all equations:

$$\forall\, i \in \{0, \cdots, R-1\}, \quad F_k^{(i)} \circ \cdots \circ F_k^{(0)} \circ \widetilde{\mathcal{A}}(x) = \widetilde{\mathcal{A}} \circ F_k^{(i)} \circ \cdots \circ F_k^{(0)}(x)$$

are satisfied. ▷

For a fixed weak key, we drew $2^{4(R-1)+6}$ plaintexts, hoping for an average of 2^6 plaintexts following the trail. We repeated the experiment for 10000 weak keys, except for $R = 7$ for which we used 6000 weak keys. Our results are depicted in Figure 5.2.

Naturally, Example 5.16 goes in the same direction as Example 5.15: as the number of rounds increases, the experimental average for the trail moves away from the theoretical one while the average for $\widetilde{\mathcal{A}} \xrightarrow{E_k} \widetilde{\mathcal{A}}$ stays much closer. What really appears in Figure 5.2, is the fact that the average probability for the trail $\widetilde{\mathcal{A}} \xrightarrow{F_k^{(0)}} \widetilde{\mathcal{A}} \to \cdots \xrightarrow{F_k^{(R-1)}} \widetilde{\mathcal{A}}$ over weak keys is not as representative as we could expect. Indeed, the probability $p = 2^{-4r}$ seems appropriate for $R = 3$, however as R grows it seems that weak keys are rather p'-weak keys where p' can take a palette of values. For the average probability of $\widetilde{\mathcal{A}} \xrightarrow{E_k} \widetilde{\mathcal{A}}$ taken over weak

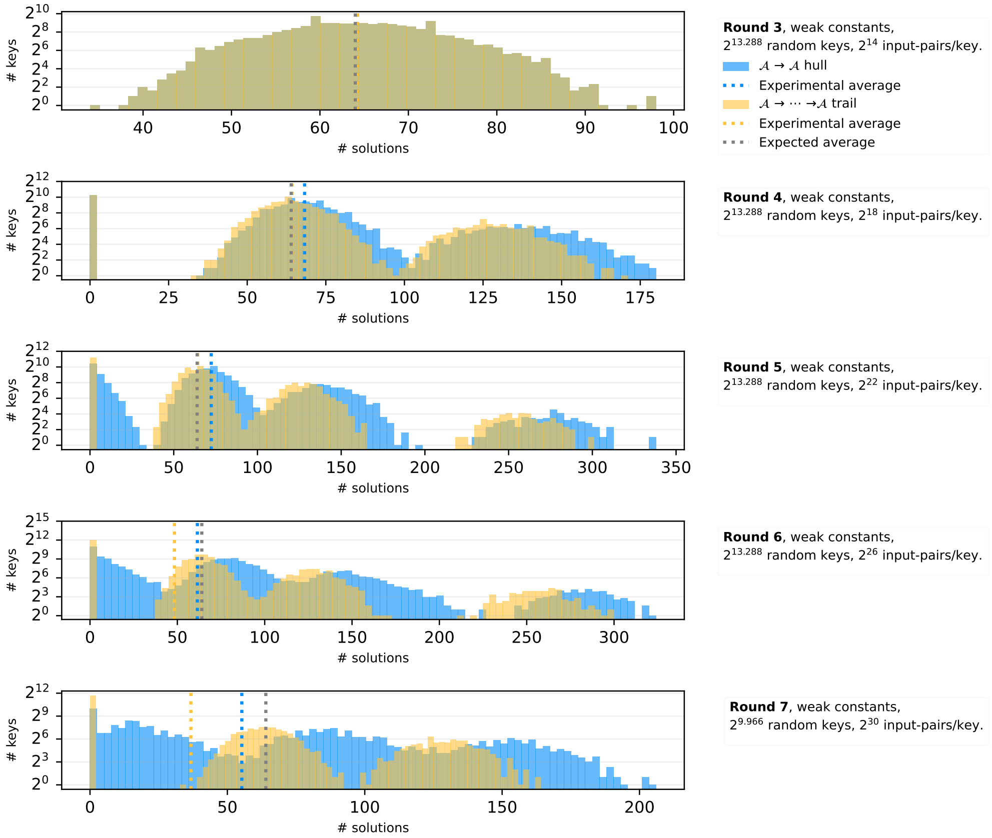


Figure 5.2: Fixed-key study: Estimation of the p-weakness through the numbers of solutions $(x, \widetilde{\mathcal{A}}(x))$ following the trail/hull. The expected average is 2^6 for every number of rounds.

keys, the distribution of p' seems to flatten as R grows and tends to a uniform distribution over $[0, 1]$. In particular, it is unclear why about half of the tested weak keys appears to be actually strong for $R \geq 4$, while some others are weaker than expected.

As just shown, the basic model seems to work well-enough to estimate the average probability for $\widetilde{\mathcal{A}} \xrightarrow{E_k} \widetilde{\mathcal{A}}$ and thus the effectiveness of this probabilistic distinguisher. However, it fails at explaining precisely the trail $\widetilde{\mathcal{A}} \xrightarrow{F_k^{(0)}} \widetilde{\mathcal{A}} \to \cdots \xrightarrow{F_k^{(R-1)}} \widetilde{\mathcal{A}}$. Explaining the observed clustering, and understanding the subclasses among the weak keys are two of the many open questions raised by our experimentation.

5.4.3 Probabilistic trails through the Sbox layer

As shown in the previous section, it is possible to build probabilistic distinguishers in the weak-key setting based on commutative cryptanalysis. While we focused before on commuting in a probabilistic way through the linear layer, it is possible to also consider probabilistic transition through the Sbox layer. When both are probabilistic, the results are *a priori* less impressive than results based on deterministic transitions over the Sbox layer. However, the probabilistic case points out some very interesting open problems. Indeed, while a designer can quickly rule out transitions $A \xrightarrow{S} B$ that hold with probability 1 by choosing an Sbox which is not self-affine equivalent, assessing almost self-affine equivalence is still a time-consuming task, especially for 8-bit Sboxes. Furthermore, it is also interesting to understand how the independence hypothesis behaves when both transitions are probabilistic.

In the case of the 4-bit Sbox of Midori, we are able to algorithmically find a pair of mappings (B, C) such that both $B \xrightarrow{S} C$ and $C \xrightarrow{S} B$ hold with probability $\frac{12}{16}$. The pair (B, C) is defined by:

$$B(x) := \begin{pmatrix} x_0 \\ x_1 + 1 \\ x_2 + 1 \\ x_0 + x_3 \end{pmatrix}, \quad \text{and} \quad C(x) := \begin{pmatrix} x_0 + x_2 + x_3 + 1 \\ x_1 + x_3 \\ x_3 \\ x_2 \end{pmatrix}$$

While the used notation is the same in Section 5.4.1.a, both pairs are distinct and should not be confused. Here, we easily observe that L_B has 8 fixed points, while L_C has 4 fixed points, which implies that the weak-key space is smaller than the one considered in Section 5.4.2.a.

We can reuse our previous probabilistic trail and replace the deterministic transitions through $\mathcal{S}$ by probabilistic ones. In that case, the probability of going through the Sbox layer is estimated as $2^{4\log_2\left(\frac{12}{16}\right)}$ as 4 Sboxes are active. The probability of going through a full round should thus be $2^{4\left(\log_2\left(\frac{12}{16}\right)-1\right)}$, and the theoretical average is computed as $2^{4\left(\log_2\left(\frac{12}{16}\right)-1\right)R+4\log_2\left(\frac{12}{16}\right)}$, because of the final round where the linear layer is omitted.

However, there is a significant divergence between our initial estimate and our experimental results and the observed probability of commuting is higher than expected. To get a better picture, we exhaustively computed the probability of having $\widetilde{\mathcal{B}} \circ M \circ \mathcal{S} = M \circ \mathcal{S} \circ \widetilde{\mathcal{C}}$, and *vice-versa* where $\widetilde{\mathcal{B}} := B \times \mathrm{Id} \times B \times \mathrm{Id}$ and $\widetilde{\mathcal{C}} := C \times \mathrm{Id} \times C \times \mathrm{Id}$. The two transitions happen with probability $2^{-5.6}$, and $2^{-8.8}$ respectively. The first one is consistent with our estimate, while the second one is not as we instead expected $2^{-9.6}$. This is a first step toward the understanding of the dependencies. Dependencies between rounds should also be expected.

5.5 Taking a step back from commutative cryptanalysis

Now that the study of commutation with affine commutants has been clearly established and illustrated, it is time to take a step back from it. To do so, we start by coming back to the distinguishers with probability one presented in Section 5.4.1, and in particular to their relationships with other classes of attacks.

5.5.1 Self-similarities and invariant subspaces

As already mentioned, because of Proposition 5.10, the distinguishers with probability 1 that we exhibited for variants of Midori64, Midori128 and for the original Scream are all affine self-similiarities. As self-similarities, they can all be interpreted either as related-key distinguishers, weak-key distinguishers, or a complementation property which reduces the effective size of the key by one bit. But because all of these self-similarities are actually coming from an iterated trail $\mathcal{A} \xrightarrow{F_k^{(0)}} \mathcal{A} \rightarrow \cdots \xrightarrow{F_k^{(R-1)}} \mathcal{A}$ which holds with probability 1 for some $\mathcal{A}$, they also imply the existence of invariant spaces. Indeed, because of Lemma 5.4, if $\mathcal{A} \circ F = F \circ \mathcal{A}$ for some bijection F, it holds that:

$$\forall\, i \in \mathbb{N}, \quad F(\mathrm{Fix}(\mathcal{A}^i)) = \mathrm{Fix}(\mathcal{A}^i),$$

and as such any $\mathrm{Fix}(\mathcal{A}^i)$ is an invariant space (under weak key assumptions) for the round function and therefore for the whole cipher.

In the case of Midori64, the mapping A that is described in Eq. (5.3) and from which $\mathcal{A}$ is derived, is an involution without fixed points. This implies that the invariant spaces described above are either empty if $i = 0$ or the full space if $i \geq 1$, both of which cannot be used as a distinguishing property. This is also the case for the commutant used in Section 5.4.1.c to build the self-similarity of Scream.

Regarding Midori128, the mapping A that is considered in Section 5.4.1.b is involutive, which means that considering $i \geq 2$ is meaningless. However A has in that case a space of fixed points V of dimension 4. This implies that V^{16} is an invariant space for the round function and the whole cipher. It corresponds exactly to the set of weak-keys as A is linear in that case so $A = L_A$ and the equality between the set of fixed points naturally follows.

Actually, this points out a slight difference between affine and linear self-similarities. In both cases, the self-similarity can always be interpreted as a weak-key distinguisher because $\mathrm{Fix}(L_{\mathcal{A}})$ is a vector subspace, and as such, is never empty. Furthermore, in the linear case because $L_{\mathcal{A}} = \mathcal{A}$, it always implies that $\mathrm{Fix}(L_{\mathcal{A}}) = \mathrm{Fix}(\mathcal{A})$ is not empty, and therefore a non-trivial invariant subspace always exists, as long as $\mathcal{A} \neq \mathrm{Id}$. In the affine case however, the set $\mathrm{Fix}(\mathcal{A})$ is either empty or an affine subspace and implies a non-trivial invariant subspace *only if* $\mathrm{Fix}(\mathcal{A})$ is not empty.

5.5.2 Differential interpretation

The case of Midori64. Another way of looking at the previous distinguishers is to consider them from a differential perspective. After all, we are studying on the one hand, $x^{(0)} \in \mathbb{F}_2^n$ and its intermediate values $x^{(r)}$ after r rounds of encryption under a key $k \in \mathbb{F}_2^\kappa$, for all $r \in [\![0, R-1]\!]$, and on the other hand $y^{(0)} := \mathcal{A}\left(x^{(0)}\right)$ and its intermediate values $y^{(r)}$ after r rounds of encryption under the key $L_{\mathcal{A}}(k)$. So we can instead study their differences $\Delta^{(r)} := x^{(r)} + y^{(r)}$ for any $r \in [\![0, R-1]\!]$. Each of them is therefore of the form $z + \mathcal{A}(z) = (\mathrm{Id} + \mathcal{A})(z)$ for some $z \in \mathbb{F}_2^n$. We then start by looking at the image set of $\mathrm{Id} + \mathcal{A}$.

In the case of Midori64, the mapping A given in Eq. (5.3) satisfies the following properties:

$$\begin{aligned} \forall x \in \langle \texttt{0x2}, \texttt{0x5}, \texttt{0x8} \rangle \,,\; x + A(x) = \texttt{0xf}, \text{ and} \\ \forall x \in \texttt{0x1} + \langle \texttt{0x2}, \texttt{0x5}, \texttt{0x8} \rangle \,,\; x + A(x) = \texttt{0xa}. \end{aligned} \tag{5.13}$$

This can indeed be immediately deduced from Eq. (5.10). We define the following two sets V and U as :

$$V := \langle \texttt{0x2}, \texttt{0x5}, \texttt{0x8} \rangle \,, \quad U := \{\texttt{0xa}, \texttt{0xf}\} \,.$$

By observing that $V \cup (\texttt{0x1} + V) = \mathbb{F}_2^4$, we deduce that $\mathrm{Im}(\mathrm{Id} + A) = U$.

As a consequence of these properties, for any $\Delta \in U^{16}$, and for any $x \in \mathbb{F}_2^n$ the value $x + \Delta$ as probability of 2^{-16} to be equal to $\mathcal{A}(x)$. Stated otherwise, a pair $(x, x + \Delta)$ coincides with the pair $(x, \mathcal{A}(x))$ with probability 2^{-16}. If this happens, and according to our commutative trail, each intermediate difference $\Delta^{(r)}$ belongs to $\mathrm{Im}(\mathrm{Id} + \mathcal{A}) = U^{16}$. We have then established that there exist 2^{16} related-key truncated differentials which each holds with a probability of at least 2^{-16}, with no heuristic argument.

To be more precise, we state this result as the following proposition and give a more detailed proof.

Proposition 5.17 (Related-key truncated differentials for Vert). *Let $\mathcal{E} = (E_k \colon \mathbb{F}_2^{64} \xrightarrow{\sim} \mathbb{F}_2^{64})_{k \in \mathbb{F}_2^{128}}$ be a member of Vert^c with $c \in V$. Let $\Delta \in U^{16}$. Let $k \in \mathbb{F}_2^{128}$. Let G^{key} be defined as in Section 5.4.1.a and let $Z^{\mathrm{diff}}(\Delta, U^{16})$ be the following set of solutions:*

$$Z^{\mathrm{diff}}(\Delta, U^{16}) := \left\{ x \in \mathbb{F}_2^{64}, E_k(x) + E_{G^{\mathrm{key}}(k)}(x + \Delta) \in U^{16} \right\}.$$

Then:

$$\left| Z^{\mathrm{diff}}(\Delta, U^{16}) \right| \geq 2^{48}.$$

Equivalently, the truncated differential $\Delta \to U^{16}$ holds in the related-key model, for any key and with probability at least 2^{-16}.

Proof. Let us consider the sets Y and Y_i for $i \in [\![0, 15]\!]$ defined by:

$$Y := (\mathrm{Id} + \mathcal{A})^{-1}(\{\Delta\}), \quad Y_i := (\mathrm{Id} + A)^{-1}(\{\Delta_i\}),$$

so that Y can be decomposed as $Y = \bigtimes_{i=0}^{15} Y_i$. The set $Z^{\text{diff}}(\Delta, U^{16})$ can of course be partitioned as:

$$Z^{\text{diff}}(\Delta, U^{16}) = \left(Z^{\text{diff}}(\Delta, U^{16}) \cap Y\right) \bigsqcup \left(Z^{\text{diff}}(\Delta, U^{16}) \setminus Y\right).$$

By construction, $x \in Y$ if and only if $x + \mathcal{A}(x) = \Delta$, *i.e.* if and only if $x + \Delta = \mathcal{A}(x)$. This implies that:

$$Z^{\text{diff}}(\Delta, U^{16}) \cap Y = \left\{x \in Y, E_k(x) + E_{G^{\text{key}}(k)}(\mathcal{A}(x)) \in U^{16}\right\}.$$

But for any $x \in \mathbb{F}_2^n$, $E_{G^{\text{key}}(k)}(\mathcal{A}(x)) = \mathcal{A}(E_k(x))$, so we get:

$$Z^{\text{diff}}(\Delta, U^{16}) \cap Y = \left\{x \in Y, (\text{Id} + \mathcal{A})(E_k(x)) \in U^{16}\right\}.$$

However, U^{16} is by definition the image of $\text{Id} + \mathcal{A}$, so we finally deduce that $Z^{\text{diff}}(\Delta, U^{16}) \cap Y = Y$. Finally, according to Eq. (5.13), any Y_i is either equal to V or $\texttt{0x1} + V$, and therefore its cardinality is always 2^3. We thus deduce that the cardinality of Y is equal to $|Y| = (2^3)^{16} = 2^{48}$ and finally conclude that:

$$\left|Z^{\text{diff}}(\Delta, U^{16})\right| \geq \left|Z^{\text{diff}}(\Delta, U^{16}) \cap Y\right| = |Y| = 2^{48}.$$

Therefore the related-key truncated differential $\Delta^{\text{in}} \rightarrow U^{16}$ holds with probability at least $2^{48-64} = 2^{-16}$.

□

This property is astonishing by the fact that its probability is very high, but also because this lower bound is *independent* of the number of rounds and holds therefore for versions with *any* number of rounds.

A direct corollary regarding differentials and not truncated differentials can also be stated.

Corollary 5.18 (Related-key differentials for Vert). *Let $\mathcal{E} = (E_k \colon \mathbb{F}_2^{64} \xrightarrow{\sim} \mathbb{F}_2^{64})_{k \in \mathbb{F}_2^{128}}$ be a member of Vert^c with $c \in V$. Let $\Delta^{\text{in}} \in U^{16}$. Let $k \in \mathbb{F}_2^{128}$. Let G^{key} be defined as in Section 5.4.1.a. Then there exists $\Delta^{\text{out}} \in U^{16}$ such that:*

$$\left|\left\{x \in \mathbb{F}_2^{64}, E_k(x) + E_{G^{\text{key}}(k)}(x + \Delta) = \Delta^{\text{out}}\right\}\right| \geq 2^{32}.$$

Equivalently, there exists $\Delta^{\text{out}} \in U^{16}$ such that $\Delta^{\text{in}} \rightarrow \Delta^{\text{out}}$ holds in the related-key model, for any key and with probability at least 2^{-32}.

Proof. According to the proof of Proposition 5.17, the 2^{48} pairs $(x, x + \Delta^{\text{in}})$, for $x \in Y$ lead to one of the 2^{16} output difference in U^{16}. A pigeonhole argument then states that at least one $\Delta^{\text{out}} \in U^{16}$ is reached by at least 2^{32} input pairs. □

If we *suppose* that the values $(\text{Id} + \mathcal{A})(E_k(x))$ with $x \in Y$ are uniformly distributed within U^{16}, then we can expect all differentials $\Delta^{\text{in}} \rightarrow \Delta^{\text{out}}$ with $\Delta^{\text{in}}, \Delta^{\text{out}} \in U^{16}$ to hold with probability 2^{-32} in the related-key model. While the two previous results are proven, this one is only stated as an assumption, supported by non-exhaustive experiments.

Assumption 5.19. *In the context described in Proposition 5.17, we expect any differential $\Delta^{\mathrm{in}} \to \Delta^{\mathrm{out}}$ with $\Delta^{\mathrm{in}}, \Delta^{\mathrm{out}} \in U^{16}$ to hold in the related-key model and for any key, with a probability close to 2^{-32}.*

Remark 5.20. Proposition 5.17, Corollary 5.18, and Assumption 5.19 can all naturally be adapted to weak-key distinguishers with the same probabilities, for the class of 2^{96} weak keys in V^{32}. ▷

The discrepancy between the expected differential behavior obtained via wide-trail arguments by the designers and our result is illustrated in Figure 5.3.

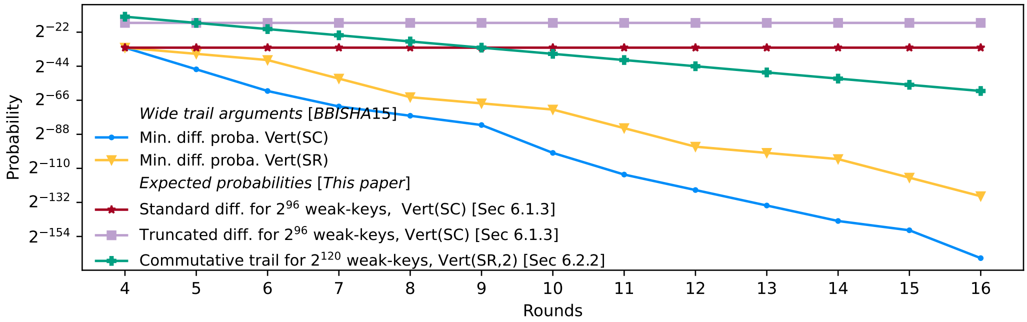


Figure 5.3: Comparison between the differential cryptanalysis of [Ban+15] and our results.

Finally, let us note that these lower bounds are only computed by considering plaintext pairs of the form $(x, x + \Delta^{\mathrm{in}})$ with $x \in Y$. For such a pair, we already noted that the same arguments prove that any intermediate difference $\Delta^{(r)}$ also belongs to U^{16}. Because 0 does not belong to U, this means that for any $i \in [\![0, 15]\!]$, ${\Delta^{(r)}}_i$ is not zero. Stated otherwise, each pair $(x, x + \Delta^{\mathrm{in}})$ with $x \in Y$ follows a differential trail which *activates all Sboxes at each round.*

In the weak-key setting, this phenomenon goes against well-established ideas. First, it is another example of how much the behavior for a fixed weak-key can deviate from the average one. Furthermore, it also proves that differentials and differentials trails activating a lot of Sboxes should be taken more into consideration. While the average (taken over independent and uniformly random round keys) probability of a trail is very small in these cases, our example either highlights how much the behavior of a trail for a weak key can deviate from the mean, and/or how much the clustering effect can be devastating. Indeed, for such a high probability for a differential $\Delta^{\mathrm{in}} \xrightarrow{E_k} \Delta^{\mathrm{out}}$ to happen, either some trails have a high probability or a lot of them contribute together, but at least one of these two events must hold.

It remains an open problem to actually understand if these phenomena are actually due to a clustering effect of many trails with small probabilities, or to the dominance of a few trails with high probability. In any case, this is even more remarkable because this holds for any number of rounds. Thinking that an increased number of rounds only leads to a stronger security is again proven wrong.

Finally, all of this is stated about a slightly-modified version of a cipher which was designed with differential resistance in mind. This once again proves how much the indubitably most studied class of attacks still remains under a cloud.

The cases of Scream and Midori128. The same actually happens for the original Scream. Indeed, with A defined in Section 5.4.1.c, the set $\mathrm{Im}(\mathrm{Id}+A) =: U$ is an affine space of dimension 3 and thus of cardinality 8. In that case, for a fixed $\Delta \in U^{16}$, a random pair $(x, x+\Delta)$ has a probability of at least $8^{-16} = 2^{-48}$ to coincide with $(x, \mathcal{A}(x))$; the corresponding truncated differential $\Delta \to U$ thus holds with probability at least 2^{-48}, independently of the number of rounds. Because $0 \notin U^{16}$, we are again assured that the contribution of 2^{-48} to the overall probability only comes from input pairs which follow trails that activate all Sboxes at all rounds.

In the case of Midori128 that is addressed in Section 5.4.1.b, or more precisely the case of Grün, $\mathrm{Im}(\mathrm{Id}+A)$ is a linear space of dimension 4 and thus of cardinality 16, which leads to truncated differentials with probability at least 2^{-64}. However in that case, as $\mathrm{Im}(\mathrm{Id}+A)$ is linear and not affine, 0 belongs to it, so the trail followed by a pair $(x, x+\Delta^{\mathrm{in}})$ with $x \in Y$ can have inactive Sboxes.

5.5.3 Ongoing studies

Finally, we mention two subjects that are still under study at the time of writing and that will be discussed in an ongoing work.

Affine uniformity. In the published paper [Bau+23], we drew the shape of a security notion related to commutative cryptanalysis, and especially to the study of Sboxes. This notion, that we call *affine uniformity*, imitates the role of differential uniformity in a differential study as a quantity needed to be minimized. However, contrary to the differential case, the considered class contains commutants with different cycle decompositions. In particular, such commutants can have a different number of fixed points. The numbers of fixed points of both A and B actually have an impact on the probability that a random function $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ and a random $x \in \mathbb{F}_2^n$ satisfy:

$$B \circ F(x) = F \circ A(x).$$

In particular, it affects the computation of the distinguishing advantage of the distinguisher, that is, the absolute difference between the expected commutative probability for the studied cryptographic function F and the probability taken over all permutations. Therefore, the distinguishing advantage can only serve as a comparative point only between pairs (A_0, B_0) and (A_1, B_1) such that $|\mathrm{Fix}(A_0)| = |\mathrm{Fix}(A_1)|$ and $|\mathrm{Fix}(B_0)| = |\mathrm{Fix}(B_1)|$. In the standard differential case, such a comparison is possible because all non-trivial translations have no fixed point.

The existence of relations between the cardinality of $Z_F^{\mathrm{comm}}(A, B)$ and the differential uniformity of F is also an open question that we are currently studying.

Key-averaged behavior. Furthermore, in order to complete the already-presented study, we are also currently looking at the average behavior of a commutative distinguisher over all possible keys. As a matter of fact, the obtained bounds show that, with independent whitening keys, a commutative (non-differential) distinguisher cannot do better in average (over all round keys) than a differential distinguisher. In other words, a commutative distinguisher can only be stronger than a differential one in a weak-key or related-key model.

5.6 Concluding remarks

Because of the closeness between the subjects addressed in this chapter and the previous one, we only add a few concluding remarks.

Commutative cryptanalysis as a general framework. It was shown in this chapter that the analysis of commutation relations, when restricted to the affine bijective case, gives rise to a rich theory which encapsulates a lot of previous techniques going from differential cryptanalysis (of conjugates) to linear self-similarities. This theory can also be effectively put in practice. Alone, it enabled us to find by a different mean the distinguisher of Vert that is presented in Chapter 4. A similar one for the block cipher Scream was also found. In that case, the distinguisher can also be interpreted as a probability-1 differential for a conjugate, because the commutant is indeed an affine involution without fixed points. Thus, to the very least, commutative cryptanalysis enables us an effective study of the differential properties of the conjugates F^G of function F, by only relying on inherent properties of F. Furthermore, for a modified version of Midori128, our framework exhibits a linear commutant, which, this time, is not a fixed-point-free involution.

A strictly larger scope? However, it remains an open question to know whether some affine commutants that are not linear nor fixed-point-free involutions can be found for a real-life cipher. This would give to commutative cryptanalysis a strictly larger scope than the one obtained by the union of linear commutants and differential analysis (of conjugates).

The probabilistic case. This is not the only open problem left with this study. The probabilistic examples given in Section 5.4.2, and especially their analysis remains unsatisfactory. They at least show the complexity of the probabilistic handling of commutative relations. In the light of the differential case that is deeply studied for decades, this is not so surprising. Nonetheless, a more profound study of probabilistic behaviors is needed. Such a study could first be done for fixed-point-free involutions. This would, at the same time, lead to a better understanding of the theories of the present and previous chapters. Furthermore, it would give examples of probabilistic differential behaviors for conjugates that we did not address previously. We could think that the differential point-of-view may help to get a better grasp to these phenomenon, but in the linear case, the toy example given by Beierle, Canteaut & Leander [BCL18] and the analysis of it made by Beyne [Bey21], seem to indicate that this intuition might not be true. In any case, all these directions would benefit from further investigations.

Differential interpretation of commutative distinguishers. Finally, as already mentioned at the end of Section 5.5.2, the differential interpretation of such commutative distinguishers, and in particular the unexpected truncated differentials that they point out definitely deserve more attention.

Chapter 6

Linear self-equivalences among known infinite APN families

This chapter is dedicated to the theoretical study of APN functions. As presented in Definition 2.26, APN functions are the functions which guarantee an optimal resistance to differential cryptanalysis. Since the definition was introduced in the early 90's [NK93], tremendous and various efforts have been made in order to better grasp these optimal objects. As of today, they still remain clouded in mystery.

As an example, only a few generic constructions of such functions are known, and connections between them are not well understood. Among them, the simpler APN monomials that are presented in Table 6.1 were the first infinite families to be discovered. Interestingly, the first proofs of APNness were actually derived from other fields of discrete mathematics, such as coding or sequence theories [CU57, Gol68, Nih72, JW93]. For quite some time, APN power mappings were believed to be the only APN functions (up to equivalence). This hypothesis was rejected around 2006, when a first isolated function [EKP06], and later a first family [BCP06] that are inequivalent to monomials were found.

However, a very important question, known as the *big APN problem*, still remains open. The problem is to determine whether or not there exist *bijective* APN functions in *even dimension* n strictly larger than 6. The first version of this problem also captured the case $n = 6$. But, Browning, Dillon, McQuistan & Wolfe [Bro+10] presented a positive answer in 2009 for this specific case. As of today, the so-called *Dillon permutation* remains the only example (up to equivalence) of APN bijection in even dimension. Its construction relies on a CCZ-equivalent function of degree 2 known as the *Kim mapping* [Bro+10] which is among the class representatives in the *Banff list* [Dil09]. This construction is now better understood

ID	Exponent	Conditions	Ref.
Gold	2^i+1	$\gcd(i,n)=1, 1 \leq i \leq \frac{n-1}{2}$	[Gol68, Nyb94, BD94]
Kasami	$2^{2i}-2^i+1$	$\gcd(i,n)=1, 1 \leq i \leq \frac{n-1}{2}$	[Kas71, JW93, Dob99b]
Welch	2^k+3	$n=2k+1$	[Dob99b]
Niho	2^k+2^r-1	$n=2k+1$ $r = \begin{cases} k/2 & \text{if } k \text{ even} \\ (3k+1)/2 & \text{if } k \text{ odd} \end{cases}$	[Nih72, Dob99a]
Inverse	$2^{2k}-1$	$n=2k+1$	[CU57, Nyb94, BD94]
Dobbertin	$2^{4k}+2^{3k}+2^{2k}+2^k-1$	$n=5k$	[Dob01]

Table 6.1: Known APN power functions over $\mathbb{F}_{2^n}$.

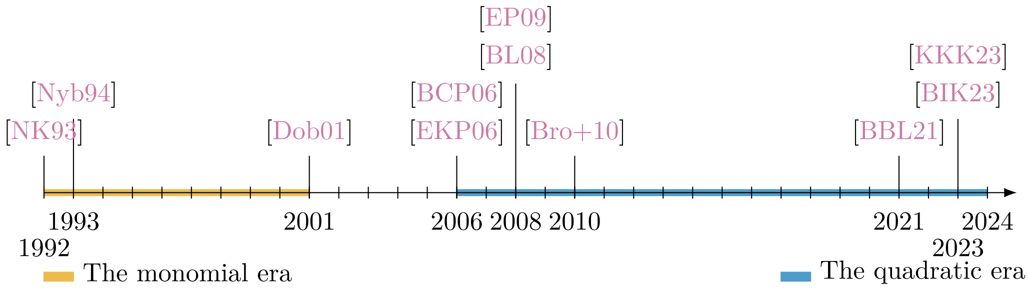


Figure 6.1: A few milestones in the study of APN functions.

thanks to the advances in the understanding of the CCZ-equivalence [CP19], and of this very specific case [PUB16]. However, the peculiarities of the Kim mapping or of the Dillon permutation are less understood.

In the last 15 years, there have been many attempts to generalize Dillon's construction by considering the CCZ-equivalence classes of simpler APN functions, typically of *quadratic* APN functions. We actually know a *single* instance (up to equivalence) which is neither equivalent to a monomial nor to a quadratic function.[1]

But looking at the properties of all known infinite families of APN functions, we have found that, despite very different representations, a vast majority of them share the same very particular structure: when they are defined over $\mathbb{F}_{2^n}$, a lot of them actually rely on the decomposition of $\mathbb{F}_{2^n}^*$ as a union of multiplicative cosets $\gamma\mathbb{F}_{2^k}^*$ of a subfield $\mathbb{F}_{2^k} \subset \mathbb{F}_{2^n}$. More precisely, they behave as a fixed power mapping, on each of the multiplicative cosets. This property was first exhibited for the Kim mapping. Indeed, the Kim mapping is defined by the following univariate form:

$$\kappa\colon \mathbb{F}_{64} \to \mathbb{F}_{64} \quad x \mapsto x^3 + x^{10} + ux^{24},$$

where u is a root of the primitive polynomial $X^6 + X^4 + X^3 + X + 1$. It was already noticed in [Bro+10] that it can be rewritten as $\kappa(x) = x^3 P(x^7)$ (where $P(x) = ux^3 + x + 1$). It follows that the Kim mapping behaves as the power mapping $x \mapsto x^3$ over the subfield $\mathbb{F}_{2^3}$ (of cardinality $7+1$). As a more general consequence, because $x \mapsto x^3$ is a bijection over $\mathbb{F}_{2^3}$, the Kim mapping satisfies for any $\gamma \in \mathbb{F}_{2^6}$, $\kappa(\gamma\mathbb{F}_{2^3}) = \kappa(\gamma)\mathbb{F}_{2^3}$, which is called the *subspace property* [Bro+10].

While most infinite APN families share a particular structure related to the multiplicative cosets of a subgroup of $\mathbb{F}_{2^n}^*$, it is surprising that this was never explicitly exhibited and studied a systematic manner. This is probably due to the different representations (univariate or multivariate) used for proving that these functions are APN, which make think that they are of different nature. The new point of view that we introduce in this chapter then provides a way to

[1] This function operates on 6 bits and is known as the Brinckmann-Leander-Edel-Pott APN cube as it was independently discovered by the first two [BL08] and last two [EP09] authors. Whether other APN functions exist outside the CCZ-equivalence classes of monomials or of quadratic functions still remains an open question.

unify many previous methodologies and definitions, while exhibiting new examples. In particular, our approach is related to the more general notion of *linear self-equivalence* [BBL21, BL22, BIK23, KKK23]. As a side effect, our work reinforces the following conjecture from [BBL21, Conjecture 1]: any APN permutation has a linearly self-equivalent CCZ-representative.

Before getting to the heart of the matter, we start by presenting the main definitions in Section 6.1. In particular, we begin from the well-known notion of homogeneity, which generalizes a property of power mappings to functions of the form $F\colon \mathbb{F}_{2^k}^\ell \to \mathbb{F}_{2^k}$. This property is involved in the definitions (or more precisely, in equivalent characterizations that we detail later) of the *cyclotomic* and *biprojectiveness* properties. They are respectively defined for functions of the form $F\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ and of the form $F\colon \mathbb{F}_{2^k}^\ell \to \mathbb{F}_{2^k}^\ell$ where $n = \ell k$. For this reason, it is hard to compare these notions, but also to relate them to properties of Boolean functions of the form $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$. Our goal is then to provide a unified point of view by relying on *linear self-equivalence.* A first step toward this objective is to precisely distinguish the cyclotomic property from the *subspace property*, which have been until now often mixed up.

In Section 6.2, we start our unification process by studying in detail the linear self-equivalences of cyclotomic or *ℓ-variate projective mappings.* To do so, we consider the linear mappings A, B involved in a linear self-equivalence relation $B \circ F \circ A = F$ of a function F, and we analyze the respective *similarity class* of A and B. The main tool at hand is a well-known *canonical form* of matrices based on the so-called *elementary divisors.* Because similarity can be studied up to isomorphisms of vector spaces, it provides a point of view well-suited to the study of functions defined over $\mathbb{F}_2^n$, $\mathbb{F}_{2^n}$ or even $\mathbb{F}_{2^k}^\ell$ with $n = \ell k$. This way, we derive three main theorems, namely Theorems 6.33, 6.37 and 6.40, which not only give a clearer view of cyclotomic mappings and ℓ-variate projective mappings, but also provide definitions which do not depend on any specific input/output space $(\mathbb{F}_2^n, \mathbb{F}_{2^n}, \mathbb{F}_{2^k}^\ell)$ nor any specific bases.

In Section 6.3, we study the known infinite families of APN functions. The whole section is dedicated to a single main result (Theorem 6.43) which states that all members of almost all of these families are linearly equivalent (and in particular CCZ-equivalent) to a cyclotomic or ℓ-variate projective mapping. Stated otherwise, despite their different initial representations (either univariate or multivariate), almost all of these families can be represented by particular linearly self-equivalent mappings. After commenting this result, its complete proof is provided.

The interest of linearly self-equivalent mappings being established in Section 6.3, we continue in Section 6.4 to study their properties. In particular, we show how much linear self-equivalence can reflect on other properties of a function. The Walsh spectrum, differential spectrum, but also in the case of quadratic APN functions, the *ortho-derivative* and its associated spectra, inherit from such symmetries. Thus, we can show how to disprove the existence of a linearly self-equivalent representative among a given equivalence class, be it a CCZ-, EA-, or linear class.

The two last sections are focused on more specific cases. In Section 6.5, after recalling some known facts about their Walsh spectra, we provide more detail about

APN cyclotomic mappings, and in particular derive some necessary conditions to be APN. We also provide explicit formula for quadratic cyclotomic and ℓ-variate projective mappings. In Section 6.6, we come back to the first example of APN cyclotomic mapping: the Kim mapping. We analyze its *Walsh zeroes* in order to understand better the fact that it is CCZ equivalent to a bijection. We conclude by regrouping the main open questions that are spread out all along the chapter.

This chapter is based on a joint work with Anne Canteaut & Léo Perrin that is under submission. A preliminary version was presented at the Thirteenth International Workshop on Coding and Cryptography (WCC 2024) [BCP24].

Contents

6.1 Cyclotomic mappings, bi-projective mappings, linearly self-equivalent mappings, and subspace property

6.1.1 Preliminaries

In this section, we recap and introduce some notation. Contrary to the other chapters of this manuscript, our variables are indexed starting for 1, and not from 0. This is intended to lighten the notation.

Let $k, \ell \geq 1$. Recall from Proposition 2.6 that any function $F\colon \mathbb{F}_{2^k}^{\ell} \to \mathbb{F}_{2^k}$ admits a unique *interpolating polynomial*, which is the unique polynomial $P \in \mathbb{F}_{2^k}[X_1, \cdots, X_\ell]$ which satisfies:

$$F(x_1, \cdots, x_\ell) = P(x_1, \cdots, x_\ell) \quad \forall\ x_1, \cdots, x_\ell \in \mathbb{F}_{2^k};$$

and which has degree $d_i \leq 2^k - 1$ in each X_i. Given $u = (u_1, \cdots, u_\ell) \in \llbracket 0, 2^k - 1 \rrbracket^{\ell}$, we denote $X^u := \prod_{i=1}^{\ell} X_i^{u_i}$.Given two sets X, Y, we denote by $\mathcal{F}(X, Y)$ the set of functions from X to Y.

The domain and codomain of a vectorial Boolean function $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$ can always be identified with other $\mathbb{F}_2$-spaces. Indeed, an $\mathbb{F}_2$-space isomorphism can always be built between two n-dimensional $\mathbb{F}_2$-spaces. In that case rather than focusing on $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$, we can instead look at $\pi_1 \circ F \circ \pi_2^{-1}$, where $\pi_1\colon \mathbb{F}_2^n \to V_1$ and $\pi_2\colon \mathbb{F}_2^m \to V_2$, where π_1, π_2 are $\mathbb{F}_2$-space isomorphims. If not stated otherwise, *isomorphism* always refers to a $\mathbb{F}_2$-linear bijection, *i.e.* a $\mathbb{F}_2$-vector space isomorphism.Handling several representations of the same functions will be a key point in our work for providing a unified point of view on several structural properties of vectorial functions.

In order to study and classify vectorial Boolean functions in an effective manner, we rely on the equivalence relations defined in Section 2.4 on page 57.

However, these equivalence relations are suited to compare functions defined from $\mathbb{F}_2^n$ to itself, but not functions two functions $F\colon \mathbb{F}_{2^k}^{\ell} \to \mathbb{F}_{2^k}^{\ell}$ and $G\colon \mathbb{F}_{2^{k'}}^{\ell'} \to \mathbb{F}_{2^{k'}}^{\ell'}$ when $(\ell, k) \neq (\ell', k')$. To do so, we define the *linear equivalence class* of a function as follows.

Definition 6.1 (Linear equivalence class)**.** Let $n = \ell k$ and F be a function from $\mathbb{F}_{2^k}^{\ell}$ to itself. Then, the linear-equivalence class of F is the subset of $\mathcal{F}(\mathbb{F}_2^n, \mathbb{F}_2^n)$

defined by:

$$\left\{\pi_1 \circ F \circ \pi_2^{-1}, \text{s.t } \pi_1, \pi_2 \colon \mathbb{F}_{2^k}^{\ell} \to \mathbb{F}_2^n \text{ are isomorphisms}\right\}.$$

▷

By definition, the linear equivalence class of *any* function is *always* a subset of $\mathcal{F}(\mathbb{F}_2^n, \mathbb{F}_2^n)$, which enables us to compare the linear-equivalence classes of functions defined over possibly different domains. This notation has the big advantage that for any isomorphisms $\psi_1, \psi_2 \colon \mathbb{F}_{2^k}^{\ell} \to \mathbb{F}_{2^{k'}}^{\ell'}$ and any function $F \colon \mathbb{F}_{2^k}^{\ell} \to \mathbb{F}_{2^k}^{\ell}$, the linear-equivalence classes of F and $\psi_1 \circ F \circ \psi_2^{-1}$ coincide. Stated otherwise, the linear equivalence class of a function is independent of the choice of bases in input and output, but also independent of the actual input or output spaces. This could be further generalized to functions $F \colon \mathbb{F}_{2^k}^{\ell} \to \mathbb{F}_{2^{k'}}^{\ell'}$ where $(\ell, k) \neq (\ell', k')$, but in our context the domain and codomain will always be equal.

6.1.2 Homogeneity

In this section, we identify connections between various concepts that were still, to the best of our knowledge, unknown. They involve a lot of different properties that appear in different contexts, using different terminologies. These properties are also defined as properties of different objects (or representations), such as Boolean functions $F \colon \mathbb{F}_2^n \to \mathbb{F}_2^n$, univariate functions $F \colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$, or multivariate functions $F \colon \mathbb{F}_{2^k}^{\ell} \to \mathbb{F}_{2^k}^{\ell}$. In most of these properties, homogeneous functions are involved, one way or another.

Definition 6.2 (Homogeneous function with exponent d)**.** Let $k, \ell, d \geq 1$ be positive integers such that $d < 2^k$. Let F be a function from $\mathbb{F}_{2^k}^{\ell}$ to $\mathbb{F}_{2^k}$. The function F is said to be *homogeneous of exponent* d if it satisfies:

$$\forall (x_1, \cdots, x_\ell) \in \mathbb{F}_{2^k}^{\ell}, \quad \forall \varphi \in \mathbb{F}_{2^k}, \quad F(x_1\varphi, \cdots, x_\ell\varphi) = \varphi^d F(x_1, \cdots, x_\ell). \tag{6.1}$$

▷

Remark 6.3. The functions defined in Definition 6.2 are sometimes known as homogeneous functions of degree d. However in our context, "degree" can already refer to the univariate, multivariate or algebraic degree of a function. We then prefer using "exponent" instead. ▷

Lemma 6.4. *Let $F \colon \mathbb{F}_{2^k}^{\ell} \to \mathbb{F}_{2^k}$ and $d < 2^k$. Then, F is homogeneous of exponent d if and only if its interpolating polynomial $P = \sum_{u \in [\![0, 2^k-1]\!]^{\ell}} a_u X^u$ satisfies:*

$$\forall u \in \left[\!\left[0, 2^k - 1\right]\!\right]^{\ell} \;\; s.t. \; \sum_{i=1}^{\ell} u_i \not\equiv d \bmod (2^k - 1), \quad a_u = 0.$$

Proof. For any $u \in [\![0, 2^k - 1]\!]^\ell$, let us denote by $\Sigma(u)$ the integer sum defined by $\Sigma(u) := \sum_{i=1}^{\ell} u_i$. Let $\varphi \in \mathbb{F}_{2^k}$. Let us introduce the following functions:

$$G\colon (x_1, \cdots, x_\ell) \mapsto F(x_1\varphi, \cdots, x_\ell\varphi), \quad H\colon (x_1, \cdots, x_\ell) \mapsto \varphi^d F(x_1, \cdots, x_\ell).$$

Then G admits $P(X_1\varphi, \cdots, X_\ell\varphi)$ as interpolating polynomial and H admits $\varphi^d P$ as interpolating polynomial. By uniqueness of the interpolating polynomial, we deduce that the two polynomials are equal: $\forall u,\ a_u\varphi^{\Sigma(u)} = a_u\varphi^d$. Choosing for φ a primitive element of $\mathbb{F}_{2^k}$, we get that, for any $a_u \neq 0$, $\varphi^{\Sigma(u)} = \varphi^d$; in other words $\Sigma(u) \equiv d \bmod 2^k - 1$. Conversely, given any φ and any u with $\Sigma(u) \equiv d \bmod 2^k - 1$, we observe that $\prod_{i=1}^{\ell}(x_i\varphi)^{u_i} = \varphi^{\Sigma(u)} \prod_{i=1}^{\ell} x_i^{u_i} = \varphi^d \prod_{i=1}^{\ell} x_i^{u_i}$, which immediately implies the result. □

Example 6.5. When $\ell = 1$, homogeneous functions are exactly the monomials functions of the form $x \mapsto cx^d, \quad c \in \mathbb{F}_{2^k}$. ▷

Example 6.6. Any homogeneous polynomial $P \in \mathbb{F}_{2^k}[X_1, \cdots, X_\ell]$ of degree d defines a homogeneous function $F\colon (\mathbb{F}_{2^n})^\ell \to \mathbb{F}_{2^n}$ of exponent d for any extension $\mathbb{F}_{2^n}$ of $\mathbb{F}_{2^k}$. However, the converse does not hold. For instance, $X_1^5X_2^2X_3^3 + X_1X_2X_3$ is not a homogeneous polynomial but still defines a homogeneous function $F\colon \mathbb{F}_8^3 \to \mathbb{F}_8$ of exponent 3 because $5 + 2 + 3 \equiv 10 \equiv 3 \bmod 7$. ▷

6.1.3 Cyclotomic mappings

This section is devoted to a particular subclass of functions from $\mathbb{F}_{2^n}$ to itself, named *cyclotomic mappings*. After studying the main properties of this family, we will show that cyclotomic mappings over $\mathbb{F}_{2^n}$ with respect to $\mathbb{F}_{2^k}^*$, where k is a divisor of n, are characterized by a multivariate representation with homogeneous coordinates. We will see in Section 6.3, that these mappings play a major role in the known infinite families of APN functions.

Definition 6.7 (Cyclotomic mapping [Wan07])**.** Let $n \geq 1$ and let $\mathbb{G} \subset \mathbb{F}_{2^n}^*$ be a multiplicative subgroup. A mapping $F\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ is a *cyclotomic mapping of exponent d with respect to* $\mathbb{G}$ if $F(0) = 0$ and:

$$\forall\ \lambda \in \mathbb{F}_{2^n},\ \exists\ c_\lambda \in \mathbb{F}_{2^n}, \forall\ x \in \mathbb{G}, \quad F(\lambda x) = c_\lambda x^d.$$

▷

Remark 6.8. For such a mapping, the original terminology introduced in [Wan07] is "cyclotomic mapping of order d and index $\frac{2^n-1}{|\mathbb{G}|}$". However, we prefer the wording of Definition 6.7 because "order" can also refer to the order of the group or of the function F, while "index" is also often overloaded. ▷

Example 6.9. When n is even, the cyclotomic mappings of exponent 0 with respect to the subgroup $\mathbb{G} = \mathbb{F}_4^*$, which is of order 3, coincide with the so-called *canonical triplicate functions* studied in [BIK23, KKK23]. More generally, when d divides $2^n - 1$, the cyclotomic mappings of exponent 0 with respect to the group

$\mathbb{G}$ of cardinality $|\mathbb{G}| = d$ coincide with the so-called *d-divisible* mappings studied in [KKK23], that is, functions that can be written as $x \mapsto P(x^d)$, for some P. ▷

Definition 6.7 equivalently means that F acts on each coset of the subgroup $\mathbb{G}$ as the *fixed* monomial function $x \mapsto x^d$, up to a multiplicative constant. This is emphasized by the following equivalent definitions.

Lemma 6.10 (Equivalent definitions). *Let $F\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ with $F(0) = 0$ and let $\mathbb{G} \subset \mathbb{F}_{2^n}^*$ be a subgroup of $\mathbb{F}_{2^n}^*$. Then, F is a cyclotomic mapping of exponent d with respect to $\mathbb{G}$ if and only if one of the following equivalent conditions holds:*

(i) $\forall \lambda \in \mathbb{F}_{2^n},\ \exists\ c_\lambda \in \mathbb{F}_{2^n}, \forall\ x \in \mathbb{G}, F(\lambda x) = c_\lambda x^d.$

(ii) $\forall \lambda \in \mathbb{F}_{2^n}, \forall\ x \in \mathbb{G}, F(\lambda x) = F(\lambda) x^d.$

(iii) *For any system Γ of representatives of $\mathbb{F}_{2^n}^*/\mathbb{G}$,*
$\forall \gamma \in \Gamma, \forall\ x \in \mathbb{G}, F(\gamma x) = F(\gamma) x^d.$

Proof. **(i) ⇔ (ii):** The fact that (i) implies (ii) is obtained by choosing $x = 1$, which leads to $c_\lambda = F(\lambda)$ for any $\lambda \in \mathbb{F}_{2^n}^*$. We then deduce that the first two definitions are equivalent.

(ii) ⇔ (iii): We only have to show that (iii) implies (ii): any $\lambda \in \mathbb{F}_{2^n}^*$ can be written $\lambda = \gamma y$ for some $\gamma \in \Gamma$ and $y \in \mathbb{G}$. Then, we deduce from (iii) that, for any $x \in \mathbb{G}$:

$$F(\lambda x) = F(\gamma x y) = F(\gamma) x^d y^d = F(\gamma y) x^d = F(\lambda) x^d\ .$$

□

Example 6.11. Let $\mathbb{F}_{2^k}$ be a subfield of $\mathbb{F}_{2^n}$. Because of the second characterization given in Lemma 6.10, we observe for instance that the trace $\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}$ relative to $\mathbb{F}_{2^k}$, as well as any $\mathbb{F}_{2^k}$-linearized polynomial, are cyclotomic mappings of exponent 1 with respect to $\mathbb{F}_{2^k}^*$. The case $d = 1, \mathbb{G} = \mathbb{F}_{2^k}^*$ is a special case of Definition 6.7 corresponding to the former and more restrictive definition of cyclotomic mappings given in [NW05]. ▷

Any function $F\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ such that $F(0) = 0$ is actually a cyclotomic mapping with respect to $\{1\}$. Therefore, we restrict ourselves to the non-trivial case where $\mathbb{G} \neq \{1\}$. Furthermore, any cyclotomic mapping with respect to $\mathbb{G}$ is also a cyclotomic mapping with respect to any subgroup of $\mathbb{G}$. We will then usually focus on the largest possible subgroup. We also notice that we can always consider $d < |\mathbb{G}|$ by replacing d by its remainder modulo $|\mathbb{G}|$.

It is also worth noting that, when the exponent d of a cyclotomic mapping F with respect to $\mathbb{G}$ is not coprime with the size of $\mathbb{G}$, then F is constant on each coset of the subgroup of order $t = \gcd(|\mathbb{G}|, d)$. This is detailed in the following definition and proposition.

Definition 6.12 (Almost t-to-1 mapping [KKK23])**.** Let $F\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ and t be a divisor of $2^n - 1$. The function F is *almost t-to-1* if there exists a unique $y_0 \in \mathrm{Im}(F)$ such that:

$$\left|F^{-1}(\{y_0\})\right| = 1, \text{ and } \quad \forall\, y \in \mathrm{Im}(F) \setminus \{y_0\}\,, \left|F^{-1}(\{y\})\right| = t.$$

▷

Proposition 6.13. *Let $\mathbb{G} \subset \mathbb{F}_{2^n}^*$ be a multiplicative subgroup of $\mathbb{F}_{2^n}^*$ and let $F\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ be a cyclotomic mapping of exponent d with respect to $\mathbb{G}$ such that $t = \gcd(d, |\mathbb{G}|) > 1$. Then, F is constant on each coset of the subgroup $\mathbb{G}' \subset \mathbb{G}$ of size t. Equivalently, F is cyclotomic of exponent 0 with respect to $\mathbb{G}'$. Most notably, if F takes distinct non-zero values on each coset of $\mathbb{G}'$, then F is almost t-to-1.*

Proof. Since t is a divisor of $|\mathbb{G}|$, there exists a subgroup $\mathbb{G}' \subset \mathbb{G}$ of size t. Then, for any $\lambda \in \mathbb{F}_{2^n}$ and any $x \in \mathbb{G}'$:

$$F(\lambda x) = F(\lambda)x^d = F(\lambda)$$

since d is a multiple of $|\mathbb{G}'|$. □

Cyclotomic mappings can also be characterized by their univariate representation, as stated in the following well-known lemma.

Lemma 6.14 (Univariate characterization)**.** *[Wan07, Lemma 1][Göl15, p.264] Let $\mathbb{G}$ be a multiplicative subgroup of $\mathbb{F}_{2^n}^*$ and $F : \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ with interpolating polynomial $P = \sum_{i=0}^{2^n-1} a_i X^i$. The mapping F is a cyclotomic mapping of exponent d with respect to $\mathbb{G}$ if and only if one of the following equivalent conditions is satisfied:*

(i) *there exists $Q \in \mathbb{F}_{2^n}[X]$ such that $P(X) = X^d Q(X^{|\mathbb{G}|})$,*

(ii) *for any $i \in [\![0, 2^n - 1]\!]$ such that $i \not\equiv d \bmod |\mathbb{G}|$, $a_i = 0$.*

Proof. The two conditions are obviously equivalent. Let $s = |\mathbb{G}|$ and $2^n - 1 = ts$. Let α be a primitive element of $\mathbb{F}_{2^n}^*$, so that α^t generates $\mathbb{G}$.

($\Longleftarrow$) Let $\lambda = \alpha^i$ and $x = \alpha^{tj} \in \mathbb{G}$. Then:

$$\begin{aligned} F(\lambda x) &= P(\alpha^{i+tj}) = \alpha^{d(i+tj)} Q(\alpha^{s(i+tj)}) \\ &= (\alpha^{tj})^d \alpha^{di} Q(\alpha^{si}) = (\alpha^{tj})^d P(\alpha^i) = x^d F(\lambda), \end{aligned}$$

where the third equality is derived from $\alpha^{st} = 1$.

($\Longrightarrow$) Conversely, let $x \in \mathbb{G}$. From Lemma 6.10, we get for any $\lambda \in \mathbb{F}_{2^n}$:

$$\sum_{i=0}^{2^n-1} a_i \lambda^i x^i = P(\lambda x) = P(\lambda)x^d = \sum_{i=0}^{2^n-1} a_i x^d \lambda^i,$$

so that $\sum_{i=0}^{2^n-1} a_i(x^d + x^i)X^i$ is the null polynomial. Therefore if $a_i \neq 0$, using a generator x of $\mathbb{G}$, we get $x^{d-i} = 1$ and thus $i \equiv d \bmod |\mathbb{G}|$.

□

The polynomials described in Lemma 6.14 are sometimes known as Wan-Lidl polynomials [WL91] and have been extensively studied, and especially in the bijective case [AW07, BPW23, CC23, Lai07, WL91, Wan17].

Example 6.15. A binomial mapping over $\mathbb{F}_{2^n}$, $x \mapsto x^i + ax^j$ with $i < j$, is a cyclotomic mapping with respect to a nontrivial subgroup $\mathbb{G} \subset \mathbb{F}_{2^n}^*$ if and only if $\gcd(j-i, 2^n-1) > 1$. Indeed, from Lemma 6.14, this equivalently means that there exists a nontrivial subgroup $\mathbb{G} \subset \mathbb{F}_{2^n}^*$ such that $i \equiv j \bmod |\mathbb{G}|$. The largest subgroup $\mathbb{G}$ for which the property holds is then the subgroup of order $\gcd(j-i, 2^n-1)$. ▷

Example 6.16. The *Kim-type mappings* defined[2] in [CL21] and also studied in [Li+21, Car15, Göl23], correspond to the mappings over $\mathbb{F}_{2^{2k}}$ with interpolating polynomials:

$$X^{3\cdot 2^k} + a_1X^{2^{k+1}+1} + a_2X^{2^k+2} + a_3X^3,\ a_1, a_2, a_3 \in \mathbb{F}_{2^{2k}}\ .$$

Since all involved exponents are equal to 3 modulo (2^k-1), these mappings are cyclotomic mappings of exponent 3 with respect to $\mathbb{F}_{2^k}^*$. As we will show later in Proposition 6.82, the interpolating polynomials of all quadratic cyclotomic mappings defined over $\mathbb{F}_{2^{2k}}$ and of exponent 3 with respect to $\mathbb{F}_{2^k}^*$ can be written as:

$$a_0X^{3\cdot 2^k} + a_1X^{2^{k+1}+1} + a_2X^{2^k+2} + a_3X^3,\ a_0, a_1, a_2, a_3 \in \mathbb{F}_{2^{2k}}\ .$$

▷

6.1.4 Cyclotomic mappings with respect to a subfield

Among all multiplicative subgroups, groups of units of subfields play a particular role. For the sake of simplicity, cyclotomic mappings with respect to the group of units of a subfield will be called *cyclotomic mappings with respect to a subfield.* For any subfield $\mathbb{F}_{2^k} \subset \mathbb{F}_{2^n}$, $\mathbb{F}_{2^n}$ can be seen as an $\mathbb{F}_{2^k}$-space of dimension $\ell := \frac{n}{k}$. In that case, a function $F : \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ can also be seen as a multivariate function, which leads to a multivariate characterization of cyclotomy.

Lemma 6.17 (Multivariate characterization)**.** *Let $n = \ell k$. Let $\pi\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^k}^\ell$ be an $\mathbb{F}_{2^k}$-linear bijection. Let $F\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ and for all $i \in [\![1, \ell]\!]$, let $F_i\colon \mathbb{F}_{2^k}^\ell \to \mathbb{F}_{2^k}$ denote the coordinates of $\pi \circ F \circ \pi^{-1}$. Then, F is a cyclotomic mapping of exponent $d < 2^k$ with respect to $\mathbb{F}_{2^k}$ if and only if, for any $i \in [\![1, \ell]\!]$, F_i is a homogeneous function of exponent d.*

Proof. Let $(b_1, \cdots, b_\ell)$ be the $\mathbb{F}_{2^k}$-basis of $\mathbb{F}_{2^n}$ corresponding to π, *i.e.* the unique basis such that $\pi(b_i)$ is the element of $\mathbb{F}_{2^k}^\ell$ having all its coordinates equal to 0

[2] The terminology "Kim-type" originates from Chase and Lisoněk [CL21], while Carlet suggests "generalized Kim" for such functions *which are also APN* [Car15].

except the i-th one, which is equal to 1. Then, the ℓ-variate coordinates $F_1, \dots, F_\ell$ of $\pi \circ F \circ \pi^{-1}$ satisfy:

$$F(\lambda) = \sum_{i=1}^{\ell} F_i(\lambda_1, \cdots, \lambda_\ell) b_i, \quad \text{where} \quad \lambda =: \sum_{i=1}^{\ell} \lambda_i b_i \text{ , with } \lambda_i \in \mathbb{F}_{2^k} \text{ for any } i.$$

$(\implies)$ By hypothesis, F satisfies: $\forall \lambda \in \mathbb{F}_{2^n}$ $\forall \varphi \in \mathbb{F}_{2^k}, F(\lambda\varphi) = F(\lambda)\varphi^d$, where equality also holds for $\varphi = 0$. Because $\varphi \in \mathbb{F}_{2^k}$, we have $\lambda\varphi = \sum_{i=1}^{\ell}(\lambda_i\varphi)b_i$. For any $i \in [\![1, \ell]\!]$, this then implies that:

$$\forall (\lambda_1, \cdots, \lambda_\ell) \in \mathbb{F}_{2^k}^{\ell} \quad \forall \varphi \in \mathbb{F}_{2^k} \quad F_i(\lambda_1\varphi, \cdots, \lambda_\ell\varphi) = \varphi^d F_i(\lambda_1, \cdots, \lambda_\ell);$$

or equivalently that all F_i are homogeneous functions of degree d.

$(\impliedby)$ Conversely, we observe that, for any $\varphi \in \mathbb{F}_{2^k}$:

$$F(\lambda\varphi) = \sum_{i=1}^{\ell} F_i(\pi(\lambda\varphi))b_i = \sum_{i=1}^{\ell} \varphi^d F_i(\pi(\lambda))b_i = \varphi^d \sum_{i=1}^{\ell} F_i(\pi(\lambda))b_i = \varphi^d F(\lambda),$$

where we use for the second equality the $\mathbb{F}_{2^k}$-linearity of π, and the homogeneity of F_i.

□

In that case, Lemma 6.17 provides an easy way to identify cyclotomic mappings through their multivariate polynomial representations. The previous characterizations of cyclotomic mapping with respect to a subfield are then summarized in the following theorem.

Theorem 6.18 (Cyclotomic mappings with respect to subfields)**.** *Let n, ℓ, k, d be positive integers such that $n = \ell k$ with $k > 1$ and $d < 2^k$. Let $F\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ with interpolating polynomial $P = \sum_{i=0}^{2^n-1} a_i X^i$. Let $F = (F_1, \cdots, F_\ell)$ be any ℓ-variate representation of F where the i-th coordinate $F_i\colon \mathbb{F}_{2^k}^{\ell} \to \mathbb{F}_{2^k}$ has $P_i = \sum_{u \in [\![0, 2^k-1]\!]^\ell} a_{u,i} X^u$ as interpolating polynomial. The following statements are equivalent:*

- *F is a cyclotomic mapping of exponent d with respect to $\mathbb{F}_{2^k}$,*
- $\forall\ \lambda \in \mathbb{F}_{2^n},\ \exists\ c_\lambda \in \mathbb{F}_{2^n},\ \forall\ \varphi \in \mathbb{F}_{2^k}, F(\lambda\varphi) = c_\lambda\varphi^d,$
- $\forall\ \lambda \in \mathbb{F}_{2^n},\ \forall\ \varphi \in \mathbb{F}_{2^k}, F(\lambda\varphi) = F(\lambda)\varphi^d,$
- *For any system Γ of representatives of $\mathbb{F}_{2^n}^*/\mathbb{F}_{2^k}^*$, $\forall \gamma \in \Gamma,\ \forall\ \varphi \in \mathbb{F}_{2^k}, F(\gamma\varphi) = F(\gamma)\varphi^d$,*
- $\exists\ Q \in \mathbb{F}_{2^n}[X], \quad P = X^d Q(X^{2^k-1}),$
- $\forall\ i \in [\![0, 2^n-1]\!]$, *such that* $i \not\equiv d \bmod 2^k - 1,\ a_i = 0,$

- $\forall\, i \in [\![1,\ell]\!]$, *$F_i$ is a homogeneous function of exponent d,*
- $\forall\, i \in [\![1,\ell]\!]\,,\ \forall\, \varphi, x_1, \cdots, x_\ell \in \mathbb{F}_{2^k}, \quad F(x_1\varphi, \cdots, x_\ell\varphi) = \varphi^d F(x_1, \cdots, x_\ell),$
- $\forall\, i \in [\![1,\ell]\!]\,,\ \forall\, u \in [\![0, 2^k-1]\!]^\ell$, *such that* $\sum_{i=1}^{\ell} u_i \not\equiv d \bmod 2^k - 1$, $a_{u,i} = 0$.

As detailed in the following definition and proposition, the so-called (q,q)-*biprojective mappings* are particular cases of cyclotomic mappings.

Definition 6.19 (Biprojective mapping [Göl22, Göl23]). Let k, q, q', r, r' be positive integers such that $k > 1$, $q = 2^r, q' = 2^{r'}$ and $r, r' < k$. A function $F \colon \mathbb{F}_{2^k}^2 \to \mathbb{F}_{2^k}^2$ with bivariate representation $F(x,y) = (F_1(x,y), F_2(x,y))$ is a (q,q')-biprojective mapping if F_1 and F_2 have interpolating polynomials of the following forms:

$$F_1(x,y) = ax^{q+1} + bx^q y + cxy^q + dy^{q+1},$$
$$F_2(x,y) = ex^{q'+1} + fx^{q'} y + gxy^{q'} + hy^{q'+1},$$

with $a, b, c, d, e, f, g, h \in \mathbb{F}_{2^k}$. ▷

Proposition 6.20 (Cyclotomic mappings and (q,q)-biprojective mappings [Göl23]). *Let $q = 2^r$. Then any (q,q)-biprojective mapping $F \colon \mathbb{F}_{2^k} \times \mathbb{F}_{2^k} \to \mathbb{F}_{2^k} \times \mathbb{F}_{2^k}$ can be expressed as $\pi \circ G \circ \pi^{-1}$, where G is a cyclotomic mapping of exponent $q+1$ with respect to $\mathbb{F}_{2^k}$ and $\pi \colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^k}^2$ is an $\mathbb{F}_{2^k}$-linear bijection.*

Proof. This is a direct corollary of the multivariate characterization of cyclotomic mappings. Indeed, we observe that any (q,q)-biprojective mapping F has homogeneous components of exponent $q+1$. By choosing an $\mathbb{F}_{2^k}$-basis (b_1, b_2) of $\mathbb{F}_{2^{2k}}$, we can build the function $G \colon \mathbb{F}_{2^{2k}} \to \mathbb{F}_{2^{2k}}$ defined by:

$$\forall x, y \in \mathbb{F}_{2^k}, \quad G(b_1 x + b_2 y) = b_1 F_1(x,y) + b_2 F_2(x,y).$$

By construction, the function G is cyclotomic of exponent $q+1$. With the mapping π defined by $\pi(b_1 x + b_2 y) = (x,y)$ for all x, y, we obtain: $F = \pi \circ G \circ \pi^{-1}$. □

Most notably, the previous proposition points out that the class of $(2,2)$-biprojective functions coincides with the family of quadratic cyclotomic mappings of exponent 3 with respect to $\mathbb{F}_{2^{\frac{n}{2}}}$ mentioned in Example 6.16. Moreover, for $q = 2^r > 2$, the (q,q)-biprojective functions correspond to the *quadratic* cyclotomic mappings of exponent (2^r+1) with respect to $\mathbb{F}_{2^{\frac{n}{2}}}$, where quadratic refers to the algebraic degree of F. Most notably, this family includes as a subclass the so-called *(closed) generalized butterflies* introduced in [PUB16], studied in [CDP17, FFW17, Li+18, CPT19], and defined by $F(x,y) = (F_1(x,y), F_1(y,x))$ with $F_1(x,y) = (x+\alpha y)^{2^r+1} + \beta y^{2^r+1}$.

6.1.5 Linearly self-equivalent mappings

When they are seen as functions from $\mathbb{F}_2^n$ to $\mathbb{F}_2^n$, cyclotomic mappings correspond to a particular subclass of linearly self-equivalent mappings. This class of mappings has been extensively studied by Beierle, Brinkmann and Leander [BBL21, BL22] in order to find new APN mappings. In particular, they observed that all known APN permutations are CCZ-equivalent to a linearly self-equivalent APN permutation and conjecture in [BBL21, Conjecture 1] that this property always holds.

In the following, given $\mathbb{F}_2$-linear bijections A_i with $i \in [\![1, \ell]\!]$ from an $\mathbb{F}_2$-space V to itself, we denote by $\mathrm{diag}(A_1, \dots, A_\ell)\colon V^\ell \to V^\ell$ the mapping defined by:

$$\forall (x_1, \dots, x_\ell) \in V^\ell, \quad \mathrm{diag}(A_1, \dots, A_\ell)(x_1, \dots, x_\ell) := (A_1(x_1), \dots, A_\ell(x_\ell)).$$

We also denote by $M_{\alpha,n}$ the multiplication mapping $x \mapsto x\alpha$ defined from $\mathbb{F}_{2^n}$ to itself.

Definition 6.21 (LE-automorphism group). [CP19, BBL21] Let $n = \ell k$, $k > 1$. The *automorphism group* of a function $F : \mathbb{F}_{2^k}^\ell \to \mathbb{F}_{2^k}^\ell$ is the set $\mathrm{Aut}(F)$ of all $\mathbb{F}_2$-affine bijections σ from $\left(\mathbb{F}_{2^k}^\ell\right)^2$ to itself such that $\{(x, F(x)), x \in \mathbb{F}_{2^k}^\ell\}$ is invariant under σ.

The *LE-automorphism group* of F is the subgroup $\mathrm{Aut}_{\mathrm{LE}}(F)$ of $\mathrm{Aut}(F)$ composed of all automorphisms of the form $\mathrm{diag}(A, B)$ for some $\mathbb{F}_2$-linear bijections $A, B\colon \mathbb{F}_{2^k}^\ell \to \mathbb{F}_{2^k}^\ell$. ▷

Definition 6.22 (Linearly self-equivalent mappings). [BBL21] A function $F : \mathbb{F}_{2^k}^\ell \to \mathbb{F}_{2^k}^\ell$ is said to be *linearly self-equivalent* if $\mathrm{Aut}_{\mathrm{LE}}(F)$ is non-trivial, *i.e.*, there exist two $\mathbb{F}_2$-linear bijections $A, B\colon \mathbb{F}_{2^k}^\ell \to \mathbb{F}_{2^k}^\ell$ with $A \neq \mathrm{Id}$ or $B \neq \mathrm{Id}$ such that $B \circ F \circ A = F$. ▷

Example 6.23. A cyclotomic mapping $F\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ of exponent d over a subgroup $\mathbb{G}$ satisfies for any $\alpha \in \mathbb{G}$:

$$M_{\alpha^{-d},n} \circ F \circ M_{\alpha,n} = F.$$

A (q, q')-biprojective function $G\colon (\mathbb{F}_{2^k})^2 \to (\mathbb{F}_{2^k})^2$ satisfies for any $\beta \in \mathbb{F}_{2^k}$:

$$\mathrm{diag}(M_{\beta^{q+1},k}, M_{\beta^{q'+1},k}) \circ G = G \circ \mathrm{diag}(M_{\beta,k}, M_{\beta,k}).$$

Both are therefore linearly self-equivalent mappings. ▷

As in the case of Definition 6.1, this definition of linear self-equivalence is compatible with any change of basis, and any change of domain. Indeed, let $F\colon \mathbb{F}_{2^k}^\ell \to \mathbb{F}_{2^k}^\ell$. Then it holds that, for all $\mathbb{F}_2$-linear bijections $\pi_1, \pi_2\colon \mathbb{F}_{2^k}^\ell \to \mathbb{F}_{2^{k'}}^{\ell'}$,

$$\begin{aligned} \mathrm{diag}(A, B) \in \mathrm{Aut}_{\mathrm{LE}}(F) &\iff B\pi_1^{-1}\pi_1 F \pi_2^{-1}\pi_2 A^{-1} = F \qquad (6.2)\\ &\iff (\pi_1 B \pi_1^{-1})(\pi_1 F \pi_2^{-1})(\pi_2 A^{-1} \pi_2^{-1}) = \pi_1 F \pi_2^{-1}\\ &\iff \mathrm{diag}\left(\pi_2 A \pi_2^{-1},\ \pi_1 B \pi_1^{-1}\right) \in \mathrm{Aut}_{\mathrm{LE}}\left(\pi_1 F \pi_2^{-1}\right). \end{aligned}$$

As a consequence of this formula, and as pointed out in [BBL21], classifying linearly self-equivalent mappings up to linear equivalence can leverage any similarity invariant of $\mathbf{GL}_n(\mathbb{F}_2)$, like the rational canonical form. We continue in this direction in Section 6.2.

Beforehand, we present another somehow-related property known as the *subspace property*, which is sometimes mistaken with cyclotomy.

6.1.6 Subspace property

The Kim mapping exhibited in [Bro+10] is a cyclotomic mapping of exponent 3 with respect to $\mathbb{F}_8$. Instead of this particular structure, Dillon *et al.* [Bro+10] highlight a more general property called the *subspace property*. In the following, we generalize it to any subgroup $\mathbb{G} \subset \mathbb{F}_{2^n}^*$ while it was originally defined in [Bro+10] for n even and $\mathbb{G} = \mathbb{F}_{2^{\frac{n}{2}}}^*$ only.

Definition 6.24 (Subspace property [Bro+10])**.** Let $F : \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ and $\mathbb{G} \subset \mathbb{F}_{2^n}^*$ be a multiplicative subgroup of $\mathbb{F}_{2^n}^*$. A mapping F satisfies the $\mathbb{G}$-*subspace property* if, for all $\lambda \in \mathbb{F}_{2^n}$, $F(\lambda\mathbb{G}) = F(\lambda)\mathbb{G}$. ▷

Because $0\mathbb{G} = \{0\}$, the definition implies that $F(\{0\}) = F(0)\mathbb{G}$, which necessarily means that $F(0) = 0$ for the cardinalities to be equal. A particular subclass of mappings satisfying the subspace property is formed by some so-called *generalized cyclotomic mappings*, which correspond to a generalization of the notion of cyclotomic mappings given in Definition 6.7. Indeed, while a cyclotomic mapping with respect to $\mathbb{G}$ acts as the same monomial mapping (up to a constant) over all cosets of $\mathbb{G}$, we may consider possibly different monomials for the different cosets, as in the following definition.

Definition 6.25 (Generalized cyclotomic mapping [BW22])**.** Let $\mathbb{G}$ be a subgroup of $\mathbb{F}_{2^n}^*$. A mapping $F\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ is called a generalized cyclotomic mapping with respect to $\mathbb{G}$ if $F(0) = 0$ and $\forall \lambda \in \mathbb{F}_{2^n}, \exists\ d_\lambda \in \mathbb{N},\ \forall\ x \in \mathbb{G}, F(\lambda x) = F(\lambda)x^{d_\lambda}$. ▷

If $F(\lambda) \neq 0$, the value of $d_\lambda \bmod |\mathbb{G}|$ only depends on the coset of λ. Indeed, it holds that for any $y, x \in \mathbb{G}$:

$$F(\lambda y)x^{d_{\lambda y}} = F(\lambda y x) = F(\lambda)y^{d_\lambda}x^{d_\lambda} = F(\lambda y)x^{d_\lambda}.$$

Therefore, as in Lemma 6.10, an equivalent condition is that, for any γ in a system of representatives of $\mathbb{F}_{2^n}^*/\mathbb{G}$, there exists $d_\gamma \in \mathbb{N}$ such that:

$$\forall x \in \mathbb{G}, \quad F(\gamma x) = F(\gamma)x^{d_\gamma}.$$

Generalized cyclotomic mappings with respect to $\mathbb{G}$ then form a subclass of the mappings satisfying the $\mathbb{G}$-subspace property if their exponents are coprime with $|\mathbb{G}|$.

Lemma 6.26. *Let $\mathbb{G}$ be a subgroup of $\mathbb{F}_{2^n}^*$ and Γ be a system of representatives of $\mathbb{F}_{2^n}^*/\mathbb{G}$. A generalized cyclotomic mapping of exponents $d_\gamma, \gamma \in \Gamma$ with respect to $\mathbb{G}$ satisfies the $\mathbb{G}$-subspace property if and only if* $\gcd(d_\gamma, |\mathbb{G}|) = 1$ *for all* $\gamma \in \Gamma$.

Proof. Let F be a generalized cyclotomic mapping with respect to $\mathbb{G}$. By definition, for any $\lambda \in \mathbb{F}_{2^n}$, we have:

$$\begin{array}{rcl} F(\lambda\mathbb{G}) & = & \{F(\lambda x), x \in \mathbb{G}\} \\ & = & \{x^{d_\lambda} F(\lambda), x \in \mathbb{G}\} \\ & = & \{x^{d_\gamma} F(\lambda), x \in \mathbb{G}\} \end{array}$$

where $\gamma \in \Gamma$ is such that $\lambda \in \gamma\mathbb{G}$. It follows that $F(\lambda\mathbb{G}) = F(\lambda)\mathbb{G}$ if and only $x \mapsto x^{d_\gamma}$ is bijective over $\mathbb{G}$, or equivalently d_γ is coprime with $|\mathbb{G}|$. □

Most notably, this points out that the subspace property as defined by Göloğlu in [Göl15], and which actually corresponds to the definition of cyclotomic mapping of exponent $(2^r + 1)$ with respect to the subfield $\mathbb{F}_{2^{n/2}}$ for any $r \geq 1$, *does not coincide* with the original subspace property recalled in Definition 6.24. Indeed, such cyclotomic mappings satisfy the $\mathbb{F}^*_{2^{n/2}}$-subspace property if and only if $\frac{n}{2\gcd(r,n/2)}$ is odd. This is not the case for instance of the APN mappings satisfying Göloğlu's subspace property when n is a multiple of 4, since the APN condition (see Proposition 6.76) implies that r is coprime with $n/2$ and contradicts Lemma 6.26.

Therefore, we want to further clarify the differences between the subspace property and the properties of (generalized) cyclotomic mappings. To this aim, we now characterize, among all mappings satisfying the $\mathbb{G}$-subspace property, the ones corresponding to generalized cyclotomic mappings with respect to $\mathbb{G}$. This characterization first requires the following proposition.

Proposition 6.27. *Let $F\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ with $F(0) = 0$, let $\mathbb{G} \subset \mathbb{F}^*_{2^n}$ be a subgroup of $\mathbb{F}_{2^n}$ and Γ be a system of representatives of $\mathbb{F}^*_{2^n}/\mathbb{G}$. Then, F has the $\mathbb{G}$-subspace property if and only if one of the following equivalent conditions is satisfied:*

(i) $\forall\lambda \in \mathbb{F}_{2^n}$, $F(\lambda\mathbb{G}) = F(\lambda)\mathbb{G}$.

(ii) $\forall\gamma \in \Gamma$, $F(\gamma\mathbb{G}) = F(\gamma)\mathbb{G}$.

(iii) $\forall\lambda \in \mathbb{F}_{2^n}$, *there exists a bijection* $G_\lambda : \mathbb{G} \to \mathbb{G}$ *such that,* $\forall x \in \mathbb{G}$, $F(\lambda x) = F(\lambda)G_\lambda(x)$.

(iv) $\forall\gamma \in \Gamma$, *there exists a bijection* $G_\gamma : \mathbb{G} \to \mathbb{G}$ *such that,* $\forall x \in \mathbb{G}$, $F(\gamma x) = F(\gamma)G_\gamma(x)$.

Proof. **(i) $\Longleftrightarrow$ (ii):** We only have to prove that (ii) implies (i). Let $\lambda \in \mathbb{F}_{2^n}$. Then, there exists $\gamma \in \Gamma$ such that $\lambda = \gamma x$. Then, $F(\lambda) \in F(\gamma)\mathbb{G}$. We then deduce that:

$$F(\lambda\mathbb{G}) = F(\gamma\mathbb{G}) = F(\gamma)\mathbb{G} = F(\lambda)\mathbb{G} \,.$$

(i) $\Longleftrightarrow$ (iii): Let $\lambda \in \mathbb{F}_{2^n}$ such that $F(\lambda) \neq 0$. We consider the mapping $G_\lambda : \mathbb{G} \to \mathbb{F}_{2^n}$ defined by:

$$G_\lambda(x) = \frac{F(\lambda x)}{F(\lambda)} \,.$$

Then, $\mathrm{Im}(G_\lambda) = \mathbb{G}$ if and only if $F(\lambda\mathbb{G}) = F(\lambda)\mathbb{G}$. Moreover, when $F(\lambda) = 0$, $F(\lambda\mathbb{G}) = \{0\}$, which means that $F(\lambda x) = F(\lambda)G_\lambda(x)$ for any bijection $G_\lambda : \mathbb{G} \to \mathbb{G}$. Therefore, we derive that (i) and (iii) (resp. (ii) and (iv)) are equivalent.

□

It is worth noticing that, when $F(\lambda) \neq 0$, all the functions $G_\lambda : \mathbb{G} \to \mathbb{G}$ in the previous definitions satisfy $G_\lambda(1) = 1$, and the same can be assumed when $F(\lambda) = 0$.

An interesting case corresponds to the situation where all functions G_λ are identical when λ varies in a coset of $\mathbb{G}$. This situation characterizes the generalized cyclotomic mappings with respect to $\mathbb{G}$ within the family of all functions satisfying the $\mathbb{G}$-subspace property.

Theorem 6.28. *Let $\mathbb{G}$ be a subgroup of $\mathbb{F}_{2^n}^*$ and Γ be a system of representatives of $\mathbb{F}_{2^n}^*/\mathbb{G}$. Let $F : \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ be a mapping satisfying the $\mathbb{G}$-subspace property, i.e., for all $\lambda \in \mathbb{F}_{2^n}$, there exists a bijection $G_\lambda : \mathbb{G} \to \mathbb{G}$ such that, $\forall x \in \mathbb{G}$, $F(\lambda x) = F(\lambda)G_\lambda(x)$. Then, for all $\gamma \in \Gamma$ and all $\lambda \in \gamma\mathbb{G}$, $G_\lambda = G_\gamma$ if and only if F is a generalized cyclotomic mapping with respect to $\mathbb{G}$ of exponents d_λ with* $\gcd(d_\lambda, |\mathbb{G}|) = 1$.

Proof. **($\Longrightarrow$)** Let us first prove that, for any F satisfying the $\mathbb{G}$-subspace property, we have that, for any $\gamma \in \Gamma$ with $F(\gamma) \neq 0$, for all $\varphi, x \in \mathbb{G}$, $G_\gamma(\varphi x) = G_\gamma(\varphi)G_{\gamma\varphi}(x)$. By definition it holds that:

$$G_\gamma(\varphi x) = \frac{F(\gamma\varphi x)}{F(\gamma)} .$$

Moreover, $F(\gamma\varphi) \neq 0$ since $F(\gamma\varphi) \in F(\gamma)\mathbb{G}$, with $F(\gamma) \neq 0$. This leads to:

$$G_\gamma(\varphi)G_{\gamma\varphi}(x) = \frac{F(\gamma\varphi)}{F(\gamma)} \times \frac{F(\gamma\varphi x)}{F(\gamma\varphi)} = \frac{F(\gamma\varphi x)}{F(\gamma)} = G_\gamma(\varphi x) .$$

By hypothesis, we know that $G_{\gamma\varphi}(x) = G_\gamma(x)$. We then deduce that, for all $\varphi, x \in \mathbb{G}$, $G_\gamma(\varphi x) = G_\gamma(\varphi)G_\gamma(x)$ This means G_γ is a multiplicative permutation of $\mathbb{G}$ with $G_\gamma(1) = 1$. Let us consider $\varphi \in \mathbb{G}$ a given generator of $\mathbb{G}$. We observe that $G_\gamma(\varphi)$ can be written as $G_\gamma(\varphi) = \varphi^{d_\gamma}$ for some d_γ. It then implies that $G_\gamma(\varphi^{d'}) = G_\gamma(\varphi)^{d'} = (\varphi^{d_\gamma})^{d'} = (\varphi^{d'})^{d_\gamma}$, so that $G_\gamma(x) = x^{d_\gamma}$ for any $x \in \mathbb{G}$. The function G_γ is therefore a power mapping and d_γ is necessarily coprime with $|\mathbb{G}|$ because it is bijective. If $F(\gamma) = 0$, then any bijection G_γ can be used, including a power permutation. We then deduce that, for any $\lambda \in \mathbb{F}_{2^n}$, $\forall x \in \mathbb{F}$, $F(\lambda x) = F(\lambda)G_\gamma(x) = F(\lambda)x^{d_\gamma}$, i.e. F is a generalized cyclotomic mapping of exponents coprime with $\mathbb{G}$.

($\Longleftarrow$) Conversely, let F be a generalized cyclotomic mapping. Then for any $\lambda \in \mathbb{F}_{2^n}$, G_λ can be defined as $G_\lambda(x) = x^{d_\lambda}$ for any $x \in \mathbb{G}$. The equality $d_\lambda \equiv d_\gamma \bmod |\mathbb{G}|$ when $\lambda \in \gamma\mathbb{G}$ is already mentioned after Definition 6.25.

Moreover, since the exponent d_γ is coprime with $|\mathbb{G}|$, $G_\lambda(x) = x^{d_\gamma}$ is a bijection on $\mathbb{G}$.

□

Theorem 6.28 enables us to have a clearer view of the situation. As a cyclotomic mapping with exponent coprime with $2^{\frac{n}{2}} - 1$, the Kim mapping appears to be a very particular case of function satisfying the $\mathbb{F}^*_{2^{\frac{n}{2}}}$-subspace property.

Contrary to cyclotomic mappings or biprojective mappings, the subspace property does not seem to imply (by definition) any kind of linear self-equivalence. For instance, let us consider the generalized cyclotomic mapping with respect to $\mathbb{F}_{2^3}$ and defined over $\mathbb{F}_{2^6}$ by:

$$F(x) = \begin{cases} x^3 & \text{if } x \in \alpha^i \mathbb{F}_{2^3} \text{ for any } i \in [\![0, 8]\!] \setminus \{1\} \\ x^5 & \text{if } x \in \alpha \mathbb{F}_{2^3} \end{cases},$$

where α is a primitive element with minimal polynomial $X^6 + X^4 + X^3 + X + 1$. It can be computationally verified that the automorphism group $\mathrm{Aut}(F)$ is trivial, and this in particular implies that this is also the case for $\mathrm{Aut}_{\mathrm{LE}}(F)$.

In the following, we rather continue studying linear self-equivalence. However, generalized cyclotomic mappings will still be mentioned in a few results in Section 6.5.2, when the generalization from the cyclotomic case is immediate.

6.2 Classification of some families of linearly self-equivalent mappings

This section is dedicated to a unified study of the cyclotomic mappings and biprojective mappings introduced in the previous section. More precisely, we study in detail the linear self-equivalences of such mappings. To do so, we consider the linear mappings A, B involved in a linear self-equivalence relation $B \circ F \circ A = F$ of a function F, and we analyze the respective *similarity class* of A and B. First, we will recall some properties of the *canonical form* of linear bijections.

6.2.1 Canonical forms of linear mappings

The family of companion matrices plays an important role when representing matrices up to similarity equivalence.

Definition 6.29 (Companion matrix)**.** Let $P(X) = X^n + \sum_{i=0}^{n-1} p_i X^i$ be a monic polynomial in $\mathbb{F}_2[X]$. Its *companion matrix* is the $n \times n$ matrix defined by:

$$C(P) = \begin{pmatrix} 0 & 0 & \cdots & 0 & p_0 \\ 1 & 0 & & \vdots & p_1 \\ 0 & 1 & \ddots & & \vdots \\ \vdots & & \ddots & 0 & p_{n-2} \\ 0 & \cdots & 0 & 1 & p_{n-1} \end{pmatrix}$$

▷

In the following, we use the canonical representation of automorphisms based on elementary divisors and which is sometimes known as the *primary rational canonical form*. This is an alternative to the one based on invariant factors (aka Frobenius normal form) and which is used in [BBL21]. Therefore, by *canonical form*, we now refer to the following well-known proposition.

Proposition 6.30 (Canonical form, elementary divisors [Her75, Page 308])**.** *Let V be an $\mathbb{F}_2$-space of dimension n. Let $A\colon V \to V$ be an $\mathbb{F}_2$-linear mapping with minimal polynomial $\prod_{i=1}^{r} P_i^{e_i}$ where $P_1, \ldots, P_r$ are distinct irreducible polynomials and all $e_i \geq 1$. Then, there exists an $\mathbb{F}_2$-basis of V in which the matrix M_A of A is of the form:*

$$M_A = \begin{pmatrix} R_1 & & \\ & \ddots & \\ & & R_r \end{pmatrix}, \textit{with } R_i = \begin{pmatrix} C\left(P_i^{e_{i,1}}\right) & & \\ & \ddots & \\ & & C\left(P_i^{e_{i,s_i}}\right) \end{pmatrix},$$

where $e_i = e_{i,1} \geq e_{i,2} \geq \ldots \geq e_{i,s_i}$ for any i. The polynomials $P_i^{e_{i,j}}$ are called the elementary divisors of A. Such a decomposition is unique, up to a reordering of the blocks.

The previous theorem is stated for a generic $\mathbb{F}_2$-space V and this is on purpose. Indeed, this enables us to handle the three main cases on which we focus on in a single stroke: functions from $\mathbb{F}_2^n$ to itself, functions from $\mathbb{F}_{2^n}$ to itself, and functions from $\mathbb{F}_{2^k}^{\ell}$ to itself with $n = \ell k$.

In the following, we denote by $\min(A)$ the *minimal polynomial* of any $\mathbb{F}_2$-linear endomorphism A. Also, the minimal polynomial of any $\alpha \in \mathbb{F}_{2^n}$ is denoted by $\min(\alpha)$.

Remark 6.31. Despite the name and notation, minimal polynomials of endomorphisms and the ones of elements of a finite field do not share all of their properties. As an example, the minimal polynomial $\min(\alpha)$ where $\alpha \in \mathbb{F}_{2^n}$ is always irreducible, while this is not the case of the minimal polynomial of a matrix. For instance, the minimal polynomial of an involutive matrix $A \neq \mathrm{Id}$ is $X^2 + 1 = (X+1)^2$. ▷

It is well-known (see for instance [MP13, pp. 311-312]) that, for any irreducible polynomial $P \in \mathbb{F}_2[X]$ of degree n, and any root $\alpha \in \mathbb{F}_{2^n}$ of P, there exists a basis of $\mathbb{F}_{2^n}$ such that the matrix of $M_{\alpha,n}$ is equal to $C(P)$. The following lemma generalizes this property, and will be very useful in our classification.

Lemma 6.32. *Let V be an $\mathbb{F}_2$-space of dimension n. Let $A\colon V \to V$ be an $\mathbb{F}_2$-linear mapping. Then the following statements are equivalent:*

(i) $\min(A)$ *is irreducible over* $\mathbb{F}_2$*,*

(ii) *there exists an irreducible polynomial $P \in \mathbb{F}_2[X]$ and an $\mathbb{F}_2$-basis in which the matrix of A is* $\mathrm{diag}(C(P), C(P), \ldots, C(P))$*,*

(iii) *there exists an irreducible polynomial $P \in \mathbb{F}_2[X]$ of degree d, $d \mid n$ such that for any root $\alpha \in \mathbb{F}_{2^n}$ of P, there exists an $\mathbb{F}_2$-linear bijection $\pi \colon V \to \mathbb{F}_{2^n}$ which satisfies: $\pi \circ A \circ \pi^{-1} = M_{\alpha,n}$.*

Proof. **(i) ⇔ (ii):** The first equivalence is a direct consequence of Proposition 6.30: if A has as unique type of block $C(P)$ for some irreducible P, this is necessarily its canonical form. Then it must hold that $\min(A) = P$ because P is the only irreducible factor of $\min(A)$ and it appears with highest power 1 in the canonical form. The minimal polynomial $\min(A)$ is therefore irreducible (and $\min(A) = P$). Conversely, if the minimal polynomial of A is irreducible, then there can be only one type of block in its canonical form, which is $C(\min(A))$.

(i & ii) $\implies$ (iii): Let d be the degree of $\min(A)$. Because of the second characterization, d is the size of the blocks, and it must then divide n. The polynomial $\min(A)$ is then irreducible of degree d, and $\mathbb{F}_{2^d} \subset \mathbb{F}_{2^n}$ is thus its splitting field. Let s be such that $n = ds$. Let $\alpha \in \mathbb{F}_{2^d}$ be a root of $\min(A)$. Let $\beta_1, \dots, \beta_s$ be an $\mathbb{F}_{2^d}$-basis of $\mathbb{F}_{2^n}$ so that any $x \in \mathbb{F}_{2^n}$ can be uniquely decomposed as $x = \sum_{i=1}^{s} x_i\beta_i$, with $x_1, \dots, x_s \in \mathbb{F}_{2^d}$. Then for any $x \in \mathbb{F}_{2^n}$ it holds that:

$$M_{\alpha,n}(x) = \alpha x = \sum_{i=1}^{s} (\alpha x_i)\beta_i = \sum_{i=1}^{s} M_{\alpha,d}(x_i)\beta_i.$$

The multiplication $M_{\alpha,n}$ is then the application of $M_{\alpha,d}$ in parallel on each coset $\beta_i\mathbb{F}_{2^d}$. But in the basis $(1, \alpha, \dots, \alpha^{d-1})$, $M_{\alpha,d}$ has $C(\min(A))$ as matrix. This means that $M_{\alpha,n}$ has $\mathrm{diag}(C(\min(A)), \dots, C(\min(A)))$ as matrix in the basis $(\alpha^i\beta_j)_{i \in [\![0,d-1]\!], j \in [\![1,s]\!]}$. By hypothesis, this is also the case of A in some basis $(v_{i,j})_{i \in [\![0,d-1]\!], j \in [\![1,s]\!]}$ of V. The linear mapping π defined by $\pi(v_{i,j}) = \alpha^i\beta_j$ for any i, j satisfies the announced property.

(i) $\impliedby$ (iii): Conversely, given P, α and π with the announced property, it holds that $\min(A) = \min(M_{\alpha,n})$. But for any $x \in \mathbb{F}_{2^n}$, it holds that:

$$P(M_{\alpha,n})x = P(\alpha)x = 0,$$

because α is a root of P. Therefore $\min(M_{\alpha,n}) \mid P$, but as P is irreducible, we deduce that $\min(M_{\alpha,n}) = P$, and thus $\min(A) = P$ is irreducible.

□

6.2.2 LE-automorphism groups of cyclotomic mappings

Using Eq. (6.2) and Lemma 6.32, we can now deduce the following correspondence between cyclotomic mappings and some linearly self-equivalent mappings.

Theorem 6.33. *Let $F : \mathbb{F}_2^n \to \mathbb{F}_2^n$ and let $\mathbb{G}$ be a subgroup of $\mathbb{F}_{2^n}^*$. Then, the following properties are equivalent.*

(i) *F belongs to the linear-equivalence class of a cyclotomic mapping with respect to $\mathbb{G}$.*

(ii) *There exists* $\mathrm{diag}(A, B) \in \mathrm{Aut}_{\mathrm{LE}}(F)$ *such that* $\min A$ *and* $\min B$ *are irreducible polynomials and* $\mathrm{ord}(A) = |\mathbb{G}|$ *and* $\mathrm{ord}(B)$ *is a divisor of* $|\mathbb{G}|$*.*

Proof. **(i)** $\Longrightarrow$ **(ii)** Let $\alpha \in \mathbb{G}$ be a generator of $\mathbb{G}$. By assumption and by Eq. (6.2) there exists an integer d and two $\mathbb{F}_2$-linear bijections $\pi_1, \pi_2 \colon \mathbb{F}_2^n \to \mathbb{F}_{2^n}$ such that $\mathrm{diag}(A, B) \in \mathrm{Aut}_{\mathrm{LE}}(F)$ where A, B are defined by:

$$A = \pi_1^{-1} \circ M_{\alpha,n} \circ \pi_1, \qquad B = \pi_2^{-1} \circ M_{\beta,n} \circ \pi_2, \tag{6.3}$$

with $\beta = \alpha^d$. By Lemma 6.32, both $\min(A)$ and $\min(B)$ are irreducible. Furthermore, A (resp. B) has the same order as $M_{\alpha,n}$ (resp. $M_{\alpha^d,n}$) which is the multiplicative order of α (resp. α^d).

(i) $\Longleftarrow$ **(ii)** Conversely, if $\min(A), \min(B)$ are irreducible, because of Lemma 6.32, they can be decomposed as in Eq. (6.3), with α such that $\langle\alpha\rangle = \mathbb{G}$ and $\mathrm{ord}(\beta) \mid |\mathbb{G}|$. This implies that $\beta \in \mathbb{G}$ and it can then be written as $\beta = \alpha^d$ for some $0 \leq d < |G|$. Then Eq. (6.2) can be used in the opposite way to deduce that $\pi_2 \circ F \circ \pi_1^{-1}$ is cyclotomic with respect to $\mathbb{G}$.

□

In other words, any function F satisfying the second condition of Theorem 6.33 admits a univariate cyclotomic representation, if the identifications between $\mathbb{F}_2^n$ and $\mathbb{F}_{2^n}$ are properly chosen.

By classifying linearly self-equivalent APN permutations according to the Frobenius normal forms of their LE-automorphisms, Beierle *et al.* [BBL21] proved that any linearly self-equivalent APN permutation in dimension 8 is CCZ-equivalent to an APN permutation with an automorphism $\mathrm{diag}(A, B)$ of one of the following two types [BBL21, Th. 4]:

1. $A = B = \mathrm{diag}(C(P), C(P))$ with $P(X) = X^4 + X^3 + X^2 + X + 1$;
2. $A = B = \mathrm{diag}(I_2, C(Q), C(Q), C(Q))$ with $Q(X) = X^2 + 1$.

A direct consequence of Theorem 6.33 is that the functions of the first type correspond to the functions in the linear-equivalence class of a cyclotomic mapping of exponent 1 with respect to the subgroup $\mathbb{G} \subset \mathbb{F}_{2^4}$ of order 5 since P is an irreducible polynomial of degree 4 and order 5. The fact that the exponent can be chosen to be 1 comes from the freedom of choice in the previous proof for α, β among all elements satisfying $\mathrm{ord}(\alpha) = \mathrm{ord}(A)$ and $\mathrm{ord}(\beta) = \mathrm{ord}(B)$. Here we can choose $\alpha = \beta$.

6.2.3 LE-automorphism groups of biprojective mappings

We have proved that the linear-equivalence classes of cyclotomic mappings are characterized by automorphisms $\mathrm{diag}(A, B)$ such that the canonical forms of A and B have all their blocks equal. Now, we focus on the class of functions such that the primary rational canonical form of B has blocks $C(P_i)$ of the same size but with possibly different minimal polynomials. This enables us to characterize the following multivariate generalization of the notion of (q, q')-biprojective functions introduced and studied by Göloğlu [Göl22, Göl23].

Definition 6.34 (ℓ-variate projective mappings)**.** Let $n = \ell k$. Let $F\colon \mathbb{F}_{2^k}^{\ell} \to \mathbb{F}_{2^k}^{\ell}$ and let $F_i\colon \mathbb{F}_{2^k}^{\ell} \to \mathbb{F}_{2^k}$, $1 \le i \le \ell$, denote its i-th coordinate. Then, F is an *ℓ-variate projective mapping* of exponents $(d_1, \dots, d_\ell)$ with respect to $\mathbb{F}_{2^k}$ if, for all i, $1 \le i \le \ell$, F_i is a homogeneous function of exponent d_i. ▷

Proposition 6.35. *Let ℓ, k, d, r, s be positive integers. Then:*

(i) *The family of ℓ-variate projective mappings of exponents $(d, \dots, d)$ coincides with the family of cyclotomic mappings of exponent d with respect to $\mathbb{F}_{2^k}$.*

(ii) *The family of 2-variate projective mappings of exponents $(2^r + 1, 2^s + 1)$ with respect to $\mathbb{F}_{2^k}$ with algebraic degree 2 coincides with the family of $(2^r, 2^s)$-biprojective mappings.*

Proof. The first item is proved in Lemma 6.17. The proof of the second item is postponed to the proof of Proposition 6.85 page 247. □

We now characterize the linear-equivalence classes of multivariate projective mappings by their LE-automorphism group. Before stating the corresponding theorem, we recall the following well-known fact.

Lemma 6.36 (Degree and order of a minimal polynomial)**.** *Let α be an element of $\mathbb{F}_{2^n}$. Then the degree of its minimal polynomial is equal to the multiplicative order of 2 modulo* $\mathrm{ord}(\alpha)$.

Proof. By definition, the degree of $\min(\alpha)$ is the number of conjugates of α. As the conjugates can be enumerated as $\alpha, \alpha^2, \alpha^{2^2} \dots$, the number of conjugates is given by the smallest $i \ge 1$ such that $\alpha^{2^i} = \alpha$, *i.e.* the smallest $i \ge 1$ such that $2^i \equiv 1 \bmod \mathrm{ord}(\alpha)$. In other words, the degree of $\min(\alpha)$ the multiplicative order of 2 modulo $\mathrm{ord}(\alpha)$. □

Theorem 6.37. *Let $n = \ell k$ and let $F : \mathbb{F}_2^n \to \mathbb{F}_2^n$. Then, the following properties are equivalent:*

(i) *F belongs to the linear-equivalence class of an ℓ-variate projective mapping of exponents $(d_1, \dots, d_\ell)$ with respect to $\mathbb{F}_{2^k}$, and for any $1 \le i \le \ell$, the multiplicative order of 2 modulo $(2^k - 1)/\gcd(d_i, 2^k - 1)$ equals k.*

(ii) *There exists* $\mathrm{diag}(A, B) \in \mathrm{Aut}_{\mathrm{LE}}(F)$ *such that* $\min(A)$ *is a primitive polynomial of degree* k *and* $\min(B)$ *is a product of distinct irreducible polynomials of degree* k*.*

Proof. **(i) $\Longrightarrow$ (ii)** By assumption, there exists $\mathrm{diag}(A, B) \in \mathrm{Aut}_{\mathrm{LE}}(F)$ such that A and B have the following forms:

$$A = \pi_1^{-1} \circ \mathrm{diag}(M_{\alpha,k}, \ldots, M_{\alpha,k}) \circ \pi_1,$$

$$B = \pi_2^{-1} \circ \mathrm{diag}(M_{\alpha^{d_1},k}, \ldots, M_{\alpha^{d_\ell},k}) \circ \pi_2,$$

where α is a primitive element of $\mathbb{F}_{2^k}$. Because of Lemma 6.32, the minimal polynomial of A is the minimal polynomial of α, and therefore a primitive polynomial of degree k. Let us denote by P_i the minimal polynomial of each α^{d_i}. By applying Lemma 6.32 to each coordinate of B, we observe that B has, as matrix representation, a diagonal matrix where the block $C(P_i)$ appears $\frac{k}{\deg(P_i)}$ times (counted with multiplicities if some $P_i = P_j$ for some $i \neq j$). But, by hypothesis and because of Lemma 6.36, the degree of P_i is equal to k, so each block $C(P_i)$ appears once (again counted with multiplicity). This then corresponds to the canonical representation of B: $\min(B)$ is therefore the least common multiple of the minimal polynomials of the blocks, which is equal to the product of the *distinct* P_i.

(ii) $\Longrightarrow$ (i) Conversely, by Lemma 6.32, any A such that $\min(A)$ is primitive and has degree k is similar to the multiplication by α where α is a generator of $\mathbb{F}_{2^k}^*$. This defines a mapping π_1. Moreover, any B such that $\min(B)$ is a product of distinct irreducible polynomials of degree k has a canonical representation of the form:

$$\mathrm{diag}(C(P_1), C(P_2), \ldots, C(P_\ell)),$$

where each P_i is an irreducible divisor of $\min(B)$. Each divisor must appear once, but some can appear several times. Therefore B is similar (for a mapping π_2) to the function $\mathrm{diag}(M_{\beta_1,k}, \ldots, M_{\beta_\ell,k})$ where each β_i is a root of P_i in $\mathbb{F}_{2^k}$. Moreover, since α is a generator of $\mathbb{F}_{2^k}^*$, any β_i can be written as α^{d_i}. This proves that $\pi_2 F \pi_1^{-1}$ is a projective mapping of exponents $(d_1, \ldots, d_\ell)$. Since P_i has degree k, k is the order of 2 modulo $\frac{(2^k-1)}{\gcd(d_i, 2^k-1)}$ by Lemma 6.36. □

When $(2^k - 1)$ is a prime number, we obtain a simpler characterization of ℓ-variate projective mappings with respect to $\mathbb{F}_{2^k}$, without any restriction on the exponents $d_1, \ldots, d_\ell$. We use that the cycle structure of a linear mapping can be derived from its canonical form, as illustrated by the following lemma.

Lemma 6.38. *Let* $A\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ *be an* $\mathbb{F}_2$*-linear mapping. Then, the following properties are equivalent:*

(i) *The cycles of A, $\sigma_A(x_0) = (x_0, Ax_0, A^2x_0, \ldots)$, have the same length for all nonzero $x_0 \in \mathbb{F}_2^n$.*

(ii) *The minimal polynomial of A is a product of distinct irreducible polynomials of the same order.*

(iii) *A has $\mathrm{diag}(C(P_1), C(P_2), \ldots, C(P_\ell))$ as canonical form where all P_i are irreducible polynomials having the same order.*

Proof. The equivalence between (ii) and (iii) is a direct consequence of the canonical form (Proposition 6.30).

(i) $\Longrightarrow$ (ii) It is well-known that, for any divisor Q of the minimal polynomial of A, there exists some $x_0 \neq 0$ such that Q is the minimal polynomial of the sequence $\sigma_A(x_0)$. We use that the period of a sequence $\sigma_A(x_0)$ with minimal polynomial $Q = P^2$ with P irreducible is $2 \times \mathrm{ord}(P)$, while the period of a sequence $\sigma_A(x_1)$ with minimal polynomial P is $\mathrm{ord}(P)$, e.g. [LN96, Theorem 8.63]. We then deduce that, if all $\sigma_A(x), x \neq 0$ have the same period, then all divisors of the minimal polynomial of A are square-free. Moreover, if the minimal polynomial of A has two irreducible divisors P_1 and P_2, then there exist x_1 and x_2 such that $\sigma_A(x_1)$ has period $\mathrm{ord}(P_1)$ and $\sigma_A(x_2)$ has period $\mathrm{ord}(P_2)$. It follows that all irreducible factors of the minimal polynomial of A have the same order.

(iii) $\Longrightarrow$ (i) Because $P_1, \ldots, P_\ell$ are irreducible of same order, they are of the same degree k by Lemma 6.36, and $\mathbb{F}_{2^k}$ is a splitting field for all of them. By hypothesis A is similar to $M = \mathrm{diag}(M_{\alpha_1,k}, \ldots, M_{\alpha_\ell,k})$, where α_i is a root of P_i. The mappings A and M share the same cycle type. But because each $M_{\alpha_i,k}$ acts independently from the others, we get that:

$$|\sigma_M(x_1, \ldots, x_\ell)| = \mathrm{lcm}\left(\left|\sigma_{M_{\alpha_1}}(x_1)\right|, \ldots, \left|\sigma_{M_{\alpha_\ell}}(x_\ell)\right|\right).$$

But for any $x, y \in \mathbb{F}_{2^k}^*$ and i, j, we get that:

$$\left|\sigma_{M_{\alpha_i}}(x)\right| = \mathrm{ord}(\alpha_i) = \mathrm{ord}(\alpha_j) = \left|\sigma_{M_{\alpha_j}}(y)\right|.$$

Therefore, whenever $(x_1, \ldots, x_\ell) \neq (0, \ldots, 0)$, its order is the common order of the elements α_i.

□

As a consequence, we can characterize the matrices having a prime order by their minimal polynomials. These matrices play an important role: as shown in [BBL21, BL22], the classification of linearly self-equivalent functions can be reduced to the classification of functions having an automorphism in $\mathrm{Aut}_{\mathrm{LE}}$ with a prime order. It can then be checked from their Frobenius normal forms that all matrices considered in [BBL21, BL22] have a minimal polynomial of the form described in the following proposition.

Proposition 6.39. *Let A be an $n \times n$-invertible matrix. Then* $\mathrm{ord}(A)$ *is an odd prime if and only if the minimal polynomial of A is of the form $(X+1)P_1(X)\dots P_\ell(X)$ or $P_1(X)\dots P_\ell(X)$ where all P_i are distinct irreducible polynomials of the same prime order $p > 2$.*

Proof. $\Longrightarrow$ If $\mathrm{ord}(A)$ is a prime p, then all cycles of A have length 1 or p. Let k' denote the dimension of the linear space composed of all fixed points of A. If $k' = 0$, then all cycles $\sigma_A(x), x \neq 0$ have the same length, implying from Lemma 6.38, that the minimal polynomial of A is a product of distinct irreducible polynomials with the same order. Assume now that $k' > 0$. Since $\mathrm{ord}(A)$ is odd, the minimal polynomial of A is not divisible by $(X+1)^2$. Then, A is similar to $A' = \mathrm{diag}(\mathrm{Id}_{k'}, C)$ where C is an $(n-k') \times (n-k')$-matrix. By observing that, for any i, $(A')^i$ is similar to $\mathrm{diag}(\mathrm{Id}_{k'}, C^i)$, we deduce that C has no nonzero fixed points and that all cycles $\sigma_C(y_0)$ for $y_0 \neq 0$ have the same length p. We deduce from Lemma 6.38 that the minimal polynomial of C can be written as the product of distinct irreducible polynomials of order $p > 2$, or equivalently that the minimal polynomial of A has the form $(X+1)P_1(X)\dots P_\ell(X)$ where all P_i are distinct irreducible polynomials of order p.

$\Longleftarrow$ We only have to consider the case where the minimal polynomial of A is of the form $(X+1)P_1(X)\dots P_\ell(X)$ where all P_i are distinct irreducible polynomials of order $p > 2$, since the other case is a direct consequence of Lemma 6.38. The canonical form of A is then $\mathrm{diag}(\mathrm{Id}_{k'}, C(P_{i_1}), \dots, C(P_{i_s}))$ where the set $\{P_{i_j}, 1 \leq j \leq s\}$ coincides with $\{P_1, \dots, P_\ell\}$ with some (possible) multiplicities. Because all P_i are irreducible and coprime with $(X+1)$, the order of A is equal to the least common multiple of the orders of all irreducible factors of $\min(A)$, which is equal to $p > 2$.

□

Theorem 6.40. *Let $F : \mathbb{F}_2^n \to \mathbb{F}_2^n$ and $k > 1$ be a divisor of n such that $(2^k - 1)$ is a prime. Assume that the span of* $\mathrm{Im}(F)$ *has dimension n. Then, the following properties are equivalent:*

(i) *F belongs to the linear-equivalence class of an ℓ-variate projective mapping with respect to $\mathbb{F}_{2^k}$.*

(ii) *There exists* $\mathrm{diag}(A, B) \in \mathrm{Aut}_{\mathrm{LE}}(F)$ *such that* $\min(A)$ *is a primitive polynomial of degree k.*

Proof. **(i)** $\Longrightarrow$ **(ii)** The proof is similar to the same result in Theorem 6.37. Indeed, the hypothesis on the orders d_i in Theorem 6.37 was used only to prove the statement about the minimal polynomial of B.

(ii) $\Longrightarrow$ **(i)** Since there exists $\mathrm{diag}(A, B) \in \mathrm{Aut}_{\mathrm{LE}}(F)$ with $\mathrm{ord}(A) = 2^k - 1$, the order of the subgroup of $\mathrm{Aut}_{\mathrm{LE}}(F)$ generated by $\mathrm{diag}(A, B)$ is a multiple of $(2^k - 1)$. Therefore, there exists $\mathrm{diag}(A', B')$ in this subgroup of order

$(2^k - 1)$. It follows that $\mathrm{lcm}(\mathrm{ord}(A'), \mathrm{ord}(B')) = 2^k - 1$ which is a prime. We deduce that either $\mathrm{ord}(A') = \mathrm{ord}(B') = 2^k - 1$, or exactly one matrix among A' and B' has order 1.

If $A' = \mathrm{Id}_n$, then $B' \circ F(x) = F(x)$ for all $x \in \mathbb{F}_{2^n}$. It follows that $\mathrm{Im}(F)$ is a subset of the set of fixed points of B', which is a vector space of dimension at most $(n-1)$ since $B' \neq \mathrm{Id}_n$. This situation is excluded by the hypotheses. If $B' = \mathrm{Id}_n$, then $F \circ A'(x) = F(x)$ where A' is a power of A. Since $\min(A)$ is a primitive polynomial of degree k, there exists an isomorphism $\pi : \mathbb{F}_{2^k}^\ell \to \mathbb{F}_2^n$, $n = k\ell$, such that $A = \pi \circ M_{\alpha,n} \circ \pi^{-1}$. Because $(2^k - 1)$ is a prime, α^s, for any $s < 2^k - 1$, is a primitive element of $\mathbb{F}_{2^k}$ too. This implies that $(\pi^{-1} \circ F \circ \pi)M_{\alpha^s} = (\pi^{-1} \circ F \circ \pi)$, i.e., $\pi^{-1} \circ F \circ \pi$ is an ℓ-variate projective mapping of orders $(0, 0, \ldots, 0)$ with respect to $\mathbb{F}_{2^k}$. Therefore, F belongs to the linear-equivalence class of an ℓ-variate projective mapping with respect to $\mathbb{F}_{2^k}$. More precisely, it is in the linear class of a cyclotomic mapping of exponent 0.

If $\mathrm{ord}(B') = 2^k - 1$ and $2^k - 1$ is an odd prime, then all cycles of B' have length 1 or $(2^k - 1)$. It follows from Proposition 6.39 that B' is similar to $B'' = \mathrm{diag}(\mathrm{Id}_{sk}, C(P_1), C(P_2), \ldots, C(P_{\ell-s}))$ where all P_i are irreducible polynomials of the same order, and therefore of the same degree k. Then, there is a function F' linearly equivalent to F such that $B'' \circ F' \circ A'' = F'$ where $A'' = \mathrm{diag}(C(P), \ldots, C(P))$ and P a primitive polynomial of degree k. This implies that there exists an isomorphism $\pi : \mathbb{F}_{2^k}^\ell \to \mathbb{F}_2^n$ such that $\pi \circ F' \circ \pi^{-1}$ is an ℓ-variate projective mapping of orders $(d_1, \ldots, d_\ell)$ with respect to $\mathbb{F}_{2^k}$ where the first s orders are zero and the other ones are determined by the roots of P_i, $1 \leq i \leq \ell - s$.

□

Just as in the case of Theorem 6.33, Theorem 6.40 enables us to determine the nature of a function $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$, from the nature of its LE automorphisms. Note that the condition on the dimension of $\langle \mathrm{Im}(F) \rangle$ is always satisfied by APN functions when $n > 2$.

Lemma 6.41 (Dimension of $\langle \mathrm{Im}(F) \rangle$ for APN functions)**.** *Let $n > 2$ and let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ be an APN function. Then $\dim(\langle \mathrm{Im}(F) \rangle) = n$.*

Proof. Let us suppose that $\dim(\langle \mathrm{Im}(F) \rangle) \leq n - 1$. In that case, and up to linear equivalence, F can be seen as a function from $\mathbb{F}_2^n$ to $\mathbb{F}_2^{n-1}$. Because F is APN, for any $\Delta^{\mathrm{in}} \in \mathbb{F}_2^n, \Delta^{\mathrm{out}} \in \mathbb{F}_2^{n-1}$, the equation $F(x + \Delta^{\mathrm{in}}) + F(x) = \Delta^{\mathrm{out}}$ must have 0 or 2 solutions x. A pigeonhole argument proves that this number is equal to 2 for all $(\Delta^{\mathrm{in}}, \Delta^{\mathrm{out}})$. Therefore, F is perfect non-linear, because $2 = 2^{n-(n-1)}$. According to Corollary 2.55, it must hold that $2(n-1) \leq n$, which is excluded because $n > 2$. □

Example 6.42 (Classes 51 & 55 of [BL22])**.** Classes 51 & 55 correspond to classes of linearly self-equivalent APN mappings over $\mathbb{F}_{2^8}$ presented in [BL22]. The functions in these classes satisfy $B \circ F \circ A = F$ for some (A, B) where A is the multiplication by an element α of order 3. By Lemma 6.32, the minimal polynomial of A is the minimal polynomial of α, *i.e.* $X^2 + X + 1$, which is of degree 2. Because $2^2 - 1 = 3$ is prime, Theorem 6.40 states that F is linearly equivalent to a 4-variate projective mapping with respect to $\mathbb{F}_4$. For Class 51, the Frobenius form of B, which is given in [BL22], is $\mathrm{diag}(\mathrm{Id}_2,\ C(X^3+1),\ C(X^3+1))$. By Proposition 6.30, the canonical form of $C(X^3 + 1)$ is $\mathrm{diag}(C(X + 1),\ C(X^2 + X + 1))$, because all irreducible divisors must appear at highest multiplicity, which is here equal to 1. Because $C(X + 1)$ is the 1×1 matrix equal to 1, B is therefore similar to:

$$\mathrm{diag}(\mathrm{Id}_2,\ \mathrm{Id}_2,\ C(X^2 + X + 1),\ C(X^2 + X + 1)).$$

This matrix is a canonical form, and by uniqueness, the one of B. Class 51 then corresponds to 4-variate projective mappings of exponent $(0, 0, 1, 1)$ with respect to $\mathbb{F}_4$. Similarly, Class 55 corresponds to 4-variate projective mappings of exponents $(0, 0, 0, 1)$ with respect to $\mathbb{F}_4$. ▷

The previous examples are (for now) sporadic examples of 4-variate APN functions. A thorough analysis of the examples coming from computational approaches such as the ones presented in [BL08, BBL21, BL22, YWL14, YP22] is left as future work. In the following, we focus on the infinite families of APN functions.

6.3 Linear self-equivalence among known infinite families of APN functions

6.3.1 Main theorem

Since we have established the relationships between the different properties considered when constructing APN functions, we can now analyse most of the infinite families of quadratic APN functions in light of the structure of their LE-automorphism groups. Most notably, while these families have been introduced with different representations (univariate or multivariate), our framework provides a unified view of these mappings which looked of very different natures at first glance. The polynomial forms of the families are presented in Tables 6.2 to 6.4. The constraints on their parameters are given in Chapter A. We prove the following theorem.

Theorem 6.43 (Infinite APN families and linear self-equivalence). *Let us consider the 19 infinite APN families listed in Tables 6.2 and 6.3. Then:*

(i) *They all contain in their linear-equivalence classes a linearly self-equivalent representative.*

(ii) *More precisely, except for Families (BCL09a/b/c) when n is odd, each family contains a cyclotomic, or a 2, 3 or 4-variate projective mapping in its linear-equivalence class.*

ID	Functions	Observations	References
(BCL08a)	$x^{2^s+1} + ax^{2^{(3-i)k+s}+2^{ik}}$	cyclotomic	[BCL08, Bud+06]
(BCL08b)	$x^{2^s+1} + ax^{2^{(4-i)k+s}+2^{ik}}$	cyclotomic	[BCL08, BCL06]
(BCV20)	$ax^{2^k+1} + x^{2^s+1} + x^{2^{s+k}+2^k} + bx^{2^{k+s}+1} + b^{2^k}x^{2^s+2^k}$	$\sim_{\text{lin}}$ biprojective	[Bra+08, BC08, BCV20]
(BCL09a)	$x^3 + a^{-1}\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_2}(a^3x^9)$	cyclotomic/($\sim_{\text{lin}}$) frob.	[BCL09a]
(BCL09b)	$x^3 + a^{-1}\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^3}}(a^3x^9 + a^6x^{18})$	cyclotomic/($\sim_{\text{lin}}$) frob.	[BCL09b]
(BCL09c)	$x^3 + a^{-1}\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^3}}(a^6x^{18} + a^{12}x^{36})$	cyclotomic/($\sim_{\text{lin}}$) frob.	[BCL09b]
(BBMM11)	$ax^{2^s+1} + a^{2^k}x^{2^{2k}+2^{k+s}} + bx^{2^{2k}+1} + ca^{2^k+1}x^{2^s+2^{k+s}}$	cyclotomic	[Bra+11a]
(BCCCV20)	$a^2x^{2^{2k+1}+1} + b^2x^{2^{k+1}+1} + ax^{2^{2k}+2} + bx^{2^k+2} + dx^3$	cyclotomic	[Bud+20]
(BHK20)	$x^3 + ax^{2^{s+i}+2^i} + a^2x^{2^{k+1}+2^k} + x^{2^{s+i+k}+2^{i+k}}$	$\sim_{\text{lin}}$ biprojective	[BHK20]
(ZKLPT22)	$a\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(bx^{2^i+1}) + a^{2^k}\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(cx^{2^s+1})$	$\sim_{\text{lin}}$ biprojective	[Zhe+22]

“frob.” refers to commutation with the Frobenius automorphism $x \mapsto x^2$.

Table 6.2: Known infinite families of univariate quadratic APN functions over $\mathbb{F}_{2^n}$. The Gold mappings are omitted.

ID	Functions	Observations	References
(ZP13)	$(x,y) \mapsto \begin{pmatrix} x^{2^s+1} + ay^{(2^s+1)2^i} \\ xy \end{pmatrix}$	$\sim_{\text{lin}}$ biprojective	[ZP13]
(T19)	$(x,y) \mapsto \begin{pmatrix} x^{2^{2s}+2^{3s}} + ax^{2^{2s}}y^{2^s} + by^{2^s+1} \\ xy \end{pmatrix}$	$\sim_{\text{lin}}$ biprojective	[Tan19]
(CBC21)	$(x,y) \mapsto \begin{pmatrix} x^{2^s+1} + x^{2^{s+k/2}}y^{2^{k/2}} + axy^{2^s} + by^{2^s+1} \\ xy \end{pmatrix}$	$\sim_{\text{lin}}$ 4-projective	[CBC21]
(G22a)	$(x,y) \mapsto \begin{pmatrix} x^{2^s+1} + xy^{2^s} + y^{2^s+1} \\ x^{2^{2s}+1} + x^{2^{2s}}y + y^{2^{2s}+1} \end{pmatrix}$	biprojective	[Göl22]
(G22b)	$(x,y) \mapsto \begin{pmatrix} x^{2^s+1} + xy^{2^s} + y^{2^s+1} \\ x^{2^{3s}}y + xy^{2^{3s}} \end{pmatrix}$	biprojective	[Göl22]
(GK21)	$(x,y) \mapsto \begin{pmatrix} x^{2^s+1} + by^{2^s+1} \\ x^{2^{s+k/2}}y + \frac{a}{b}xy^{2^{s+k/2}} \end{pmatrix}$	biprojective	[GK21]
(CLV22a)	$(x,y) \mapsto \begin{pmatrix} x^{2^s+1} + xy^{2^s} + ay^{2^s+1} \\ x^{2^{2s}+1} + ax^{2^{2s}}y + (1+a)^{2^s}xy^{2^{2s}} + ay^{2^{2s}+1} \end{pmatrix}$	biprojective	[CLV22]
(LK23a)	$(x,y,z) \mapsto \begin{pmatrix} x^{2^s+1} + x^{2^s}z + yz^{2^s} \\ x^{2^s}z + y^{2^s+1} \\ xy^{2^s} + y^{2^s}z + z^{2^s+1} \end{pmatrix}$	3-projective $\sim_{\text{lin}}$ cyclotomic	[LK23]
(LK23b)	$(x,y,z) \mapsto \begin{pmatrix} x^{2^s+1} + xy^{2^s} + yz^{2^s} \\ xy^{2^s} + z^{2^s+1} \\ x^{2^s}z + y^{2^s+1} + y^{2^s}z \end{pmatrix}$	3-projective $\sim_{\text{lin}}$ cyclotomic	[LK23]

Table 6.3: Known infinite families of bivariate or trivariate quadratic APN functions over $\mathbb{F}_{2^n}$

(iii) *When n is odd, any function of (BCL09a/b/c) is linearly-equivalent to a function which commutes with the Frobenius automorphism $x \mapsto x^2$.*

Finally, all (APN) power mappings are cyclotomic and commute with the Frobenius automorphism.

A lot of subcases were already pointed out in several previous papers such as [Car11, BBL21, CBC21, Göl22, GK21, BIK23, KKK23]. In particular, and to the best of our knowledge, Carlet first pointed out in [Car11, Theorem 1] the relevance of studying functions of the form:

$$\begin{aligned} F(x,y) &= \left(xy, a_1 x^{2^i+2^j} + b_1 x^{2^i} y^{2^j} + c_1 x^{2^j} y^{2^i} + d_1 y^{2^i+2^j}\right) \\ &= \left(xy, (a_2 x^{2^{j-i}+1} + b_2 x y^{2^{j-i}} + c_2 x^{2^{j-i}} y + d_2 y^{2^{j-i}+1})^{2^i}\right), \end{aligned} \tag{6.4}$$

that is, functions that are linearly equivalent to a $(2, 2^{j-i})$-projective mapping. Carlet also proved in [Car11, Section 4.2.1] that previously known infinite families, namely the ones given in [Bra+08, BC08], and that are today included in the (BCV20) family [BCV20], fall within this category. As pointed out by Example 6.9, the works [GK21, BIK23] present proofs of cyclotomy of exponent 0 with respect to $\mathbb{F}_4$ for a lot of these families. In the following, we generalize them into cyclotomy or (bi-)projectiveness proofs over larger groups. Unlike these works however, we make (almost) no distinction between even or odd values for n.

We believe that such a general observation deserves to be in the spotlight. We therefore prove all the cases and give credit to authors of previous works (that we know of) in the proof. The proof is postponed to the following section. We first present a few observations about this result.

Remark 6.44. Theorem 6.43 mentions representatives in the linear equivalence classes, but all the representatives presented in the proof actually lie in an $\mathbb{F}_{2^k}$-linear equivalence class with $k > 1$. Furthermore, this is not an exhaustive result, and some functions of these families have linearly self-equivalent representatives of several types. Examples of this situation are presented in Remark 6.48. ▷

The following informal corollary of Theorem 6.43 raises many open questions.

Corollary 6.45 ((Informal) Infinite APN families and self-equivalence)**.** *Almost all infinite families of APN functions have linearly self-equivalent representatives in their linear-equivalence class, whose LE-automorphism group contains (A, B) where A, B are either $\mathbb{F}_{2^k}$-linear with $k > 1$ with very particular minimal polynomials characterized in Theorem 6.37, or where both A and B coincide with the Frobenius automorphism.*

This observation is rather surprising. Indeed, from theoretical arguments, what we (for now) know is that any quadratic function $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ is always *EA* self-equivalent, but *a priori* not linearly self-equivalent. Indeed, for any $\Delta \in \mathbb{F}_2^n$, there exists an affine mapping $A_\Delta\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ such that that for any $x \in \mathbb{F}_2^n$:

$$F(x + \Delta) + F(x) = A_\Delta(x).$$

ID	Functions	Obs.	Ref.
(CLV22b)	$(x,y) \mapsto \begin{pmatrix} x^3+xy+xy^2+ay^3 \\ x^5+xy+ax^2y^2+ax^4y+(1+a)^2xy^4+ay^5 \end{pmatrix}$	?	[CLV22]
(LZLQ22b)	$(x,y) \mapsto \begin{pmatrix} x^3+xy^2+y^3+xy \\ x^5+x^4y+y^5+xy+x^2y^2 \end{pmatrix}$	?	[Li+22]
(LZLQ22a)	$L(x)^{2^k+1}+bx^{2^k+1}$	?	[Li+22]

Table 6.4: Remaining infinite families to classify.

In other words, F satisfies:

$$\begin{pmatrix} T_\Delta & 0 \\ A_\Delta & \mathrm{Id} \end{pmatrix} \in \mathrm{Aut}(F).$$

This is for instance proven in [Bra+11b, Proposition 1]. This observation can be extended to *extended-linear* self-equivalent if $F(0)=0$, see e.g [KZ21, Proposition 2. 2].

Problem 6.46. *Does the property described in Corollary 6.45 hold for the three families in Table 6.4? For the sporadic APN functions such as those in [BL08, BBL21, BL22, YWL14, YP22]?*

We show for instance in Example 6.62 below, that the Brickmann-Leander-Edel-Pott [BL08, EP09] cubic for $n=6$ cannot be represented as a cyclotomic mapping nor as an ℓ-variate projective mapping. More generally, and in line with [BBL21, Conjecture 1], we raise the following open problem.

Problem 6.47. *Does the CCZ-equivalence class of any APN function contain a linearly self-equivalent mapping?*

Theorem 6.43 then unifies (almost) all the research directions followed to search for infinite APN families. Answering the question asked in Problem 6.47 would enable us to understand whether these directions are direct generalizations extrapolated from the monomial case, or whether they correspond to an inherent property of APN mappings. The specific cases highlighted in Problem 6.46 could help address this problem or give some clues toward a definite answer.

6.3.2 Proof of Theorem 6.43

This section is dedicated to the proof of Theorem 6.43. We proceed case by case and start with the most obvious ones.

Power mapping. A power mapping is a cyclotomic mapping of exponent d with respect to $\mathbb{F}_{2^n}$ and it obviously commutes with $x \mapsto x^2$.

Multivariate Families. Among the bivariate and trivariate families given in Table 6.3, we directly observe from their polynomial forms that:

- (G22a) is $(2^s+1,\ 2^{2s}+1)$-projective,
- (G22b) is $(2^s+1,\ 2^{3s}+1)$-projective,
- (GK21) is $(2^s+1,\ 2^{2+\frac{k}{2}}+1)$-projective,
- (CLV22a) is $(2^s+1,\ 2^{2s}+1)$-projective,
- (LK23a) and (LK23b) are $(2^s+1,\ 2^s+1,\ 2^s+1)$-projective,

all of them being ℓ-variate projective mappings by construction. Note that Theorem 6.18 shows that (LK23a) and (LK23b) have a representative which is a cyclotomic mapping of exponent 2^s+1 with respect to $\mathbb{F}_{2^k}$ in their linear-equivalence class. Furthermore, the families (ZP13), (T19) and the polynomials defined by Eq. (6.4) and introduced by Carlet have been proven linearly-equivalent to biprojective mappings by Göloğlu [Göl22]. More precisely:

- for (ZP13), using the ($\mathbb{F}_{2^k}$-linear) mapping $L\colon (x,y) \mapsto (x, y^{2^{k-i}})$, we find a linear-equivalent function $F \circ L$ which is a $(2^s+1,\ 2^{k-i}+1)$-projective mapping,
- for (T19), using the ($\mathbb{F}_{2^k}$-linear) mapping $L\colon (x,y) \mapsto (x^{2^{k-2s}}, y)$, we find a linear-equivalent function $F \circ L$ which is a $(2^s+1,\ 2^{k-2s}+1)$-projective mapping,
- for the polynomials of Eq. (6.4), using the ($\mathbb{F}_{2^k}$-linear) mapping $L\colon (x,y) \mapsto (x,\ y^{2^{k-i}})$, we find a linear-equivalent function $L \circ F$ which is a $(2,\ 2^{i-j}+1)$-projective mapping.

(CBC21). As we can observe the first coordinate of this mapping has monomials of degree d where $d \equiv 2^s+1 \bmod 2^{\frac{k}{2}}-1$, but not modulo 2^k-1. When substituting each monomial with $x \leftarrow a+\zeta b$, $y \leftarrow c+\zeta d$, with $\zeta \in \mathbb{F}_{2^k} \backslash \mathbb{F}_{2^{k/2}}$, and $a,b,c,d \in \mathbb{F}_{2^{k/2}}$, we observe that the obtained monomials in a,b,c,d are all of degree 2^s+1, because $a^{2^{k/2}} = a$ and the same holds for b,c,d. The same holds for the second coordinate. Therefore, the functions of (CBC21) are linearly equivalent to $(2^s+1,\ 2^s+1,\ 2,\ 2)$-projective mappings.

Let us now focus on the univariate families.

(BCL09a/b/c). First of all the families (BCL09a), (BCL09b) (BCL09c) were for instance identified as canonical triplicates when n is even in [BIK23], and their image set was studied in [KKK23]. When n is even, they correspond to cyclotomic mappings. More precisely:

- when n is even, (BCL09a), (BCL09b), (BCL09c) are made of cyclotomic mappings of exponent 0 with respect to $\mathbb{F}_4$, because they can be written as $P(x^3)$,
- when n is odd, it is observed in the original paper [BCL09b, Section II.B] that, when a takes different values, all the obtained functions within a fixed family (BCL09a), (BCL09b) or (BCL09c) are linearly-equivalent. We can actually focus on the case $a = 1$, by using $x \mapsto a^{\frac{1}{3}}x$ as a change of variables. In that case, the corresponding function has all its coefficients in $\mathbb{F}_2$, and therefore commutes with the Frobenius automorphism.

(BCL08a/b). Non-trivial linear self-equivalences were identified for Families (BCL08a) and (BCL08b) in [BBL21, Examples 2 & 3]. They can be reinterpreted as proofs of cyclotomy. Indeed, let us look at the difference between both exponents modulo $2^k - 1$. We observe that:

$$(2^{(3-i)k+s} + 2^{ik}) - (2^s + 1) \equiv 2^s + 1 - 2^s - 1 \equiv 0 \bmod 2^k - 1.$$

From Theorem 6.18, Family (BCL08a) is a family of cyclotomic mappings of exponent $2^s + 1$ with respect to $\mathbb{F}_{2^k}$, where $n = 3k$. Similarly, we obtain:

$$(2^{(4-i)k+s} + 2^{ik}) - (2^s + 1) \equiv 2^s + 1 - 2^s - 1 \equiv 0 \bmod 2^k - 1.$$

Therefore, Family (BCL08b) is a family of cyclotomic mappings of exponent $2^s + 1$ with respect to $\mathbb{F}_{2^k}$, where $n = 4k$.

(BCCCV20) & (BBMM11). If we look at Family (BCCCV20), we observe that the monomials appearing in the polynomials are:

$$x^{2^{2k+1}+1},\ x^{2^{k+1}+1},\ x^{2^{2k}+2},\ x^{2^k+2},\ \text{and } x^3\ ,$$

so that its exponents are all equal to 3 modulo $2^k - 1$. This implies that the family consists exclusively of cyclotomic mappings of exponent 3 with respect to $\mathbb{F}_{2^k}$, where $n = 3k$. Regarding Family (BBMM11), the same applies, but we need to take into account some of the constraints on the parameters. Since $n = 3k$, we can look at cyclotomy with respect to $\mathbb{F}_{2^3}$. First, we reduce all the exponents modulo $2^3 - 1$ and obtain in that case: $2^s + 1,\ 2^{2k} + 1,\ 2^{2k} + 1,\ 2^s + 1$ because $k + s \equiv 0 \bmod 3$ by construction. Furthermore, again by construction, we have that $\gcd(3, k) = 1$, which implies $k \not\equiv 0 \bmod 3$. From these two constraints, we deduce that either $k \equiv 1$ and $s \equiv 2$, or, $k \equiv 2$ and $s \equiv 1$. In any case, it holds that $s \equiv 2k \bmod 3$. This proves that all exponents are equal modulo $2^3 - 1$, and the family then consists of cyclotomic mappings of exponent $2^{s'} + 1 \in \{3, 5\}$ with respect to $\mathbb{F}_{2^3}$ where s' is the remainder of s modulo 3.

(ZKLPT22). This family lies among bivariate families as well. Indeed, by definition, $a \notin \mathbb{F}_{2^k}^*$ (see Table A.2), so (a, a^{2^k}) is an $\mathbb{F}_{2^k}$-basis of $\mathbb{F}_{2^n}$ where $n = 2k$. We observe that for any $\varphi \in \mathbb{F}_{2^k}, x \in \mathbb{F}_{2^n}$, we then have:

$$\begin{aligned} F(\varphi x) &= a\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}\left(b(\varphi x)^{2^i+1}\right) + a^{2^k}\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}\left(c(\varphi x)^{2^s+1}\right) \\ &= a\varphi^{2^i+1}\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}\left(bx^{2^i+1}\right) + a^{2^k}\varphi^{2^s+1}\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}\left(cx^{2^s+1}\right), \end{aligned}$$

because $\varphi^{2^i+1}, \varphi^{2^s+1} \in \mathbb{F}_{2^k}$. It is then linearly equivalent to a $(2^i+1,\ 2^s+1)$-projective mapping.

(BCV20). Let F be the function defined by $F(x) = ax^{2^k+1} + x^{2^s+1} + x^{2^{s+k}+2^k} + bx^{2^{k+s}+1} + b^{2^k}x^{2^s+2^k}$ where $n = 2k$ for some $k \geq 1$. Let $\alpha \in \mathbb{F}_{2^n}$ be a primitive element, and let us consider the $\mathbb{F}_{2^k}$-basis $(1, \alpha)$ and its dual basis (β_1, β_α) which satisfies:

$$\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\beta_1 \cdot 1) = 1, \mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\beta_1 \cdot \alpha) = 0, \mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\beta_\alpha \cdot 1) = 0, \mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\beta_\alpha \cdot \alpha) = 1.$$

In particular, we observe that $\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\beta_\alpha) = 0$, in other words, it holds that $\beta_\alpha = \beta_\alpha^{2^k}$, or stated otherwise that $\beta_\alpha \in \mathbb{F}_{2^k}$. Let $\gamma \in \mathbb{F}_{2^n}$, and let us focus on $\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\gamma F)$. Let $x \in \mathbb{F}_{2^n}$. Then it holds that:

$$\begin{aligned} \mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\gamma F(x)) =& \gamma(ax^{2^k+1} + x^{2^s+1} + x^{2^{s+k}+2^k} + bx^{2^{k+s}+1} + b^{2^k}x^{2^s+2^k}) + \\ & \gamma^{2^k}(ax^{2^k+1} + x^{2^s+1} + x^{2^{s+k}+2^k} + bx^{2^{k+s}+1} + b^{2^k}x^{2^s+2^k})^{2^k} \\ =& \mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\gamma a)x^{2^k+1} + \lambda x^{2^s+1} + \lambda x^{2^{s+k}+2^k} + b\lambda x^{2^{s+k}+1} + b^{2^k}\lambda x^{2^s+2^k}, \end{aligned}$$

where $\lambda = \mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\gamma)$. This simply comes from the fact that $x^{2^{2k}} = x$. We can therefore express the two coordinates of F with respect to the $\mathbb{F}_{2^k}$-basis $(1, \alpha)$ as :

$$\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\beta_1 F(x)) = \mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\beta_1 a)x^{2^k+1} + x^{2^s+1} + x^{2^{s+k}+2^k} + bx^{2^{s+k}+1} + b^{2^k}x^{2^s+2^k},$$

because $\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\beta_1) = 1$, but also:

$$\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\beta_\alpha F(x)) = \mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\beta_\alpha a)x^{2^k+1},$$

because $\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\beta_\alpha) = 0$. Let us introduce the linear bijection $L\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ that is defined by:

$$\forall x, y \in \mathbb{F}, \quad L(x + \alpha y) = \left(x + \frac{\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\beta_1 a)}{\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\beta_\alpha a)}y\right) + \alpha y.$$

By construction, $a \notin \mathbb{F}_{2^k}$ (see Table A.1), but as $\beta_\alpha \in \mathbb{F}_{2^k}$, we deduce that $\beta_\alpha a \notin \mathbb{F}_{2^k}$, and therefore $\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\beta_\alpha a) \neq 0$, so that L is well-defined. We then observe that:

$$\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\beta_1 \cdot L \circ F(x)) = x^{2^s+1} + x^{2^{s+k}+2^k} + bx^{2^{s+k}+1} + b^{2^k}x^{2^s+2^k}, \text{and}$$

$$\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\beta_\alpha \cdot L \circ F(x)) = \beta_\alpha \mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(a)x^{2^k+1}.$$

In particular, the bivariate terms that can appear in the 1-coordinate of $L \circ F$ are terms of degree $2^s + 1$ because all exponents e of its univariate monomials satisfy $e \equiv 2^s + 1 \bmod 2^k - 1$. Similarly, the bivariate terms that can appear in the α-coordinate are terms of degree 2. Therefore $L \circ F$ is a $(2^s + 1, 2)$ biprojective APN function.

(BHK20). Let $F\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ be defined by $F(x) = x^3 + ax^{2^{s+i}+2^i} + a^2x^{2^{k+1}+2^k} + x^{2^{s+i+k}+2^{i+k}}$ where $n = 2k$ for some $k \geq 1$. We proceed in a manner similar to the previous proof, except that we use the fact that a is by definition an element of order 3, so $a^2 = a^{-1}$, and also that k is odd, see Table A.2. Let $\gamma \in \mathbb{F}_{2^n}$, $x \in \mathbb{F}_{2^n}$. Then it holds that:

$$\begin{aligned}
\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\gamma F(x)) &= \gamma(x^3 + ax^{2^{s+i}+2^i} + a^2x^{2^{k+1}+2^k} + x^{2^{s+i+k}+2^{i+k}}) + \\
&\quad \gamma^{2^k}(x^3 + ax^{2^{s+i}+2^i} + a^2x^{2^{k+1}+2^k} + x^{2^{s+i+k}+2^{i+k}})^{2^k} \\
&= (\gamma + \gamma^{2^k}a)x^3 + (\gamma a + \gamma^{2^k})x^{2^{s+i}+2^i} + \\
&\quad (\gamma a^{-1} + \gamma^{2^k})x^{2^{k+1}+2^k} + (\gamma + \gamma^{2^k}a^{-1})x^{2^{s+i+k}+2^{i+k}} \\
&= (\gamma + \gamma^{2^k}a)(x^3 + a^{-1}x^{2^{k+1}+2^k}) + \\
&\quad (\gamma a + \gamma^{2^k})(x^{2^{s+i}+2^i} + a^{-1}x^{2^{s+i+k}+2^{i+k}})
\end{aligned}$$

In particular, the terms x^3 and $x^{2^{k+1}+2^k}$ appear if and only if $\gamma + \gamma^{2^k}a \neq 0$. Stated otherwise, if $\gamma \neq 0$, both terms do not appear if and only if $\gamma^{2^k-1} = a^{-1}$, *i.e.* if and only if γ is a $(2^k - 1)$-th root of a^{-1}. Such a root exists. Indeed, if β is a primitive element of $\mathbb{F}_{2^n}^*$, then $\beta^{\frac{2^n-1}{3}}$ is a generator of $\mathbb{F}_{2^2}^*$ and it can be rewritten as $\beta^{\frac{2^n-1}{3}} = (\beta^{\frac{2^k+1}{3}})^{2^k-1}$. In particular, because k is odd, $\frac{2^k+1}{3}$ is an integer, so this precisely states that $\beta^{\frac{2^k+1}{3}}$ or $(\beta^{\frac{2^k+1}{3}})^2$ is a $(2^k - 1)$-th root of a^{-1}. Similarly, the terms $x^{2^{s+i}+2^i}$ and $x^{2^{s+i+k}+2^{i+k}}$ do not appear in $\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\gamma F)$ if and only if γ is a $(2^k - 1)$-th root of a. Finally, $(\beta^{\frac{2^k+1}{3}}, \beta^{\frac{2(2^k+1)}{3}})$ is an $\mathbb{F}_{2^k}$-basis of $\mathbb{F}_{2^n}$, because $\frac{\beta^{\frac{2(2^k+1)}{3}}}{\beta^{\frac{2^k+1}{3}}} = \beta^{\frac{2^k+1}{3}} \notin \mathbb{F}_{2^k}$. In this basis, the coordinates are homogeneous of exponent 3 and $2^i(2^s + 1)$ respectively, because the monomials in each coordinate are of degree equal to 3 and $2^i(2^s + 1)$ modulo $2^k - 1$.

This concludes our proof. □

Remark 6.48. As already mentioned, some functions in these classes have multiple linearly self-equivalent representatives of different natures. For example, a single function can be at the same time linearly-equivalent to a cyclotomic mapping, but also to a function which commutes with the Frobenius automorphism. It can also happen that a single representative has two types of linear self-equivalence. For instance as mentioned in [BCL09b, Section II.B], when n is even Family (BCL09a) can be split into two distinct linear classes. Indeed all functions are either linearly-equivalent to the function with $a = 1$, or to the one where a is a fixed primitive element of $\mathbb{F}_{2^n}$. In the first case, the representative with $a = 1$ is cyclotomic, but it also commutes with the Frobenius automorphism, because[3] its coefficients are in $\mathbb{F}_2$. The same also holds when n is even for Families (BCL09b), (BCL09c).

Another example can be derived from (BBMM11). We showed that this class is composed of cyclotomic mappings of exponent 3 or 5 with respect to $\mathbb{F}_{2^3}$. However, if $b = c = 0$, then only two terms remain and their exponents are both equal to $2^s + 1$ modulo $2^k - 1$. These specific functions are therefore cyclotomic with respect to $\mathbb{F}_{2^3}$, but also with respect to $\mathbb{F}_{2^k}$.

Finally, for Family (BCV20), on top of the biprojective property, according to the results reported in [BIK23, Section 7], the functions were computationally proven linearly-equivalent to cyclotomic mappings of exponent 0 with respect to $\mathbb{F}_4$, up to dimension 12. ▷

6.4 Properties of mappings having a linearly self-equivalent mappings

In the previous section, we pointed out the importance of linear self-equivalence, especially for the study of APN functions. We highlight in this section some properties which are consequences of the existence of a linearly self-equivalent mapping within the linear-equivalence class of a function. We first present how the symmetries inherent to this pattern can be captured by other means than the polynomial representation.

6.4.1 Image set and Walsh spectrum of linearly self-equivalent mappings

If a function $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ is linearly self-equivalent, then its image set is very constrained. For instance, some properties of F can be derived from the cycle structures of the involved linear mappings. A first trivial property is the following one.

Proposition 6.49. *Let $F : \mathbb{F}_2^n \to \mathbb{F}_2^n$ and $\mathrm{diag}(A, B) \in \mathrm{Aut}_{\mathrm{LE}}(F)$. Then, the image set of F can be partitioned into cycles of B. Most notably, the image set of F is invariant under B.*

Proof. It holds that: $F(\mathbb{F}_2^n) = B \circ F \circ A^{-1}(\mathbb{F}_2^n) = B(F(\mathbb{F}_2^n))$, so $F(\mathbb{F}_2^n)$ is invariant under B. This precisely states that $F(\mathbb{F}_2^n)$ is a disjoint union of cycles of B. □

This result does not bring any new information in the case where F is bijective, but is helpful when F is not bijective, which is the case of most known APN

[3] The functions commuting with $x \mapsto x^2$ are precisely the functions whose coefficients are in $\mathbb{F}_2$. Note that APN functions of this specific form are classified up to dimension 9 in [Yu+20].

functions. In the following, we highlight how linear self-equivalence can be captured as a property of the *Walsh transform* of the function F.

As noted at the end of Section 2.3.4.d, we adapt the definition of the Walsh transform to functions of the form $G\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ (resp. $H\colon \mathbb{F}_{2^k}^{\ell} \to \mathbb{F}_{2^k}^{\ell}$), so that the Walsh coefficients can be enumerated using $\alpha, \beta \in \mathbb{F}_{2^n}$ (resp. $\alpha, \beta \in \mathbb{F}_{2^k}^{\ell}$), instead of $\alpha, \beta \in \mathbb{F}_2^n$. In that case, it suffices to replace in Definition 2.15 the standard dot product by a scalar product defined over $\mathbb{F}_{2^n}$ (resp. $\mathbb{F}_{2^k}^{\ell}$). Over $\mathbb{F}_{2^n}$, we can then consider the scalar product defined by:

$$\forall\ x, y \in \mathbb{F}_{2^n}, \quad x \cdot y = \mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_2}(xy). \tag{6.5}$$

Over $\mathbb{F}_{2^k}^{\ell}$, we use the one defined by:

$$\forall\ z, t \in \mathbb{F}_{2^n}, \quad z \cdot t = (z_1, \dots, z_\ell) \cdot (t_1, \dots, t_\ell) = \sum_{i=1}^{\ell} \mathrm{Tr}_{\mathbb{F}_{2^k}/\mathbb{F}_2}(z_i t_i). \tag{6.6}$$

We also denote by A^* the *adjoint operator* of a linear mapping A, for a given scalar product, *i.e.*, the linear mapping such that,

$$x \cdot A(y) = A^*(x) \cdot y,\ \forall x, y\ .$$

Because the Walsh coefficients of a mapping $B \circ F \circ A$ in the linear-equivalence class of F are in one-to-one correspondence with the Walsh coefficients of F (see Proposition 2.64, page 59), this implies that linear self-equivalence is captured by some spectral symmetries.

Lemma 6.50 (Spectral characterization of linear self-equivalence)**.** *Let A, B be bijective linear mappings from $\mathbb{F}_2^n$ to itself. Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$. Then $B \circ F \circ A = F$ if and only if:*

$$\forall \alpha \in \mathbb{F}_2^n, \forall \beta \in \mathbb{F}_2^n, \quad W_F\left((A^{-1})^*(\alpha), B^*(\beta)\right) = W_F(\alpha, \beta).$$

Proof. The Walsh coefficient of the left-hand side is precisely the Walsh coefficient of $B \circ F \circ A$ in (α, β). It is then a consequence of the fact that two functions are equal if and only if their Walsh transforms are equal. □

Corollary 6.51. *Let A, B be bijective linear mappings from $\mathbb{F}_2^n$ to itself. Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ be such that $B \circ F \circ A = F$. Let $\mathcal{L}$ be the function from $\mathbb{F}_2^n \times \mathbb{F}_2^n$ to itself that is defined by:*

$$\forall\ x, y \in \mathbb{F}_2^n, \quad \mathcal{L}(x, y) = \left((A^{-1})^*(x), B^*(y)\right)$$

Assume that the lengths of the cycles $\sigma_{\mathcal{L}}(x_0, y_0)$, for all nonzero $(x_0, y_0) \in \mathbb{F}_2^n \times \mathbb{F}_2^n$, are divisible by L. Then each value v in the multiset

$$\{\!\{W_F(\alpha, \beta),\ \text{s.t. } \alpha, \beta \in \mathbb{F}_{2^n},\ (\alpha, \beta) \neq (0, 0)\}\!\}$$

appears $L \cdot t_v$ times for some $t_v \geq 1$. In that case, the greatest common divisor of the numbers of occurrences of the values in the Walsh spectrum is divisible by L.

Proof. Let $(\alpha, \beta) \in \mathbb{F}_2^n \times \mathbb{F}_2^n$ be such that $(\alpha, \beta) \neq (0,0)$. By assumption, the multiset $\{\!\{W_F(\mathcal{L}^i(\alpha,\beta)), i \in [\![0, L-1]\!]\}\!\}$ contains the single value $W_F(\alpha, \beta)$ with multiplicity λL. Indeed, because $\sigma_{\mathcal{L}}(\alpha,\beta)$ is of length λL, this value corresponds to λL distinct Walsh coefficients. The divisibility is then an immediate consequence of the fact that the multiset $\{\!\{W_F(\alpha,\beta),\ s.t.\ (\alpha,\beta) \neq (0,0)\}\!\}$ can be partitioned according to the decomposition of $\mathcal{L}$ into cycles with disjoint supports. □

Corollary 6.52 (Walsh coefficients of a cyclotomic mapping)**.** *Let $F \colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ be a cyclotomic mapping of exponent d with respect to $\mathbb{G} \subset \mathbb{F}_{2^n}^*$. Then:*

$$\forall \alpha, \beta \in \mathbb{F}_{2^n}, \forall x \in \mathbb{G}, \quad W_F(\alpha, \beta x^d) = W_F(\alpha x^{-1}, \beta).$$

Furthermore, the greatest common divisor of the numbers of occurrences of the values in the Walsh spectrum is divisible by $\frac{|\mathbb{G}|}{\gcd(d,|\mathbb{G}|)}$ for $d > 0$, and by $|\mathbb{G}|$ when $d = 0$.

Proof. In the case of a cyclotomic mapping of exponent d with respect to $\mathbb{G}$ we can choose $A^{-1} = M_{x,n}$ where $x \in \mathbb{G}$ and $B = M_{x^d,n}$. The relation between the Walsh coefficients is then a direct consequence of Lemma 6.50. Moreover, the cycle decomposition of $(A^{-1})^*$ (resp. of B^*) is the same as the cycle decomposition of A (resp. of B). Starting from a nonzero element, all cycles of A have length $\mathrm{ord}(x)$, and all cycles of B have length $\mathrm{ord}(x^d)$. If $d \neq 0$, then

$$\mathrm{ord}(x^d) = \frac{\mathrm{ord}(x)}{\gcd(d, \mathrm{ord}(x))}\ .$$

Most notably, we deduce the result by choosing for x a generator of $\mathbb{G}$. When $d = 0$, all cycles of $\mathcal{L} = ((A^{-1})^*(x), y)$ have length $|\mathbb{G}|$. □

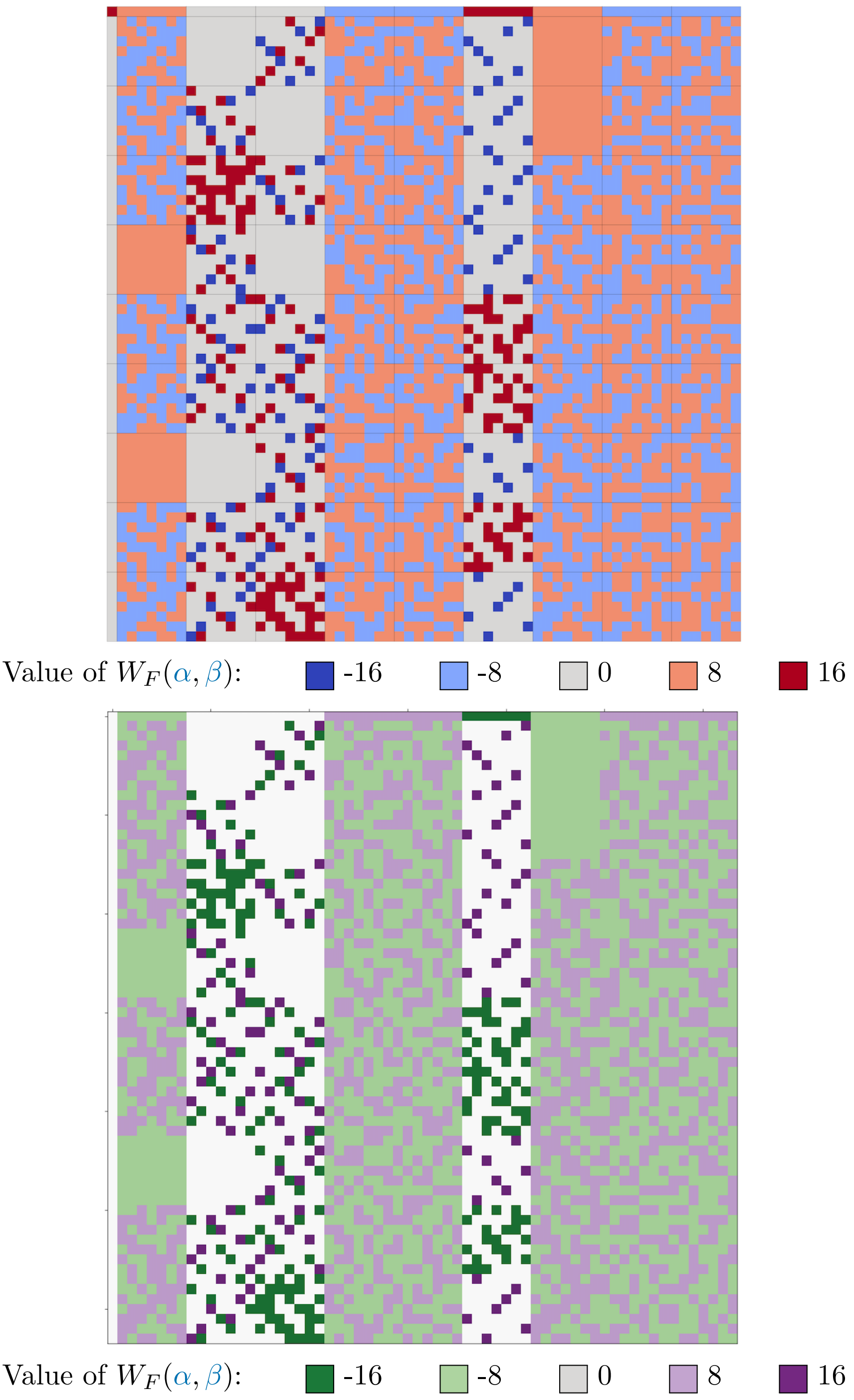
The following corollary can be proved in a similar manner.

Corollary 6.53 (Walsh coefficients of an ℓ-variate projective mapping)**.** *Let $F \colon \mathbb{F}_{2^k}^\ell \to \mathbb{F}_{2^k}^\ell$ be an ℓ-variate projective mapping of exponents $(d_1, \ldots, d_\ell)$ with respect to $\mathbb{F}_{2^k}$. Then:*

$$\forall \alpha, \beta \in \mathbb{F}_{2^k}^\ell, \forall x \in \mathbb{F}_{2^k}^*,\ W_F\left(\alpha, \left(\beta_1 x^{d_1}, \ldots, \beta_\ell x^{d_\ell}\right)\right) = W_F\left(\left(\alpha_1 x^{-1}, \ldots, \alpha_\ell x^{-1}\right), \beta\right).$$

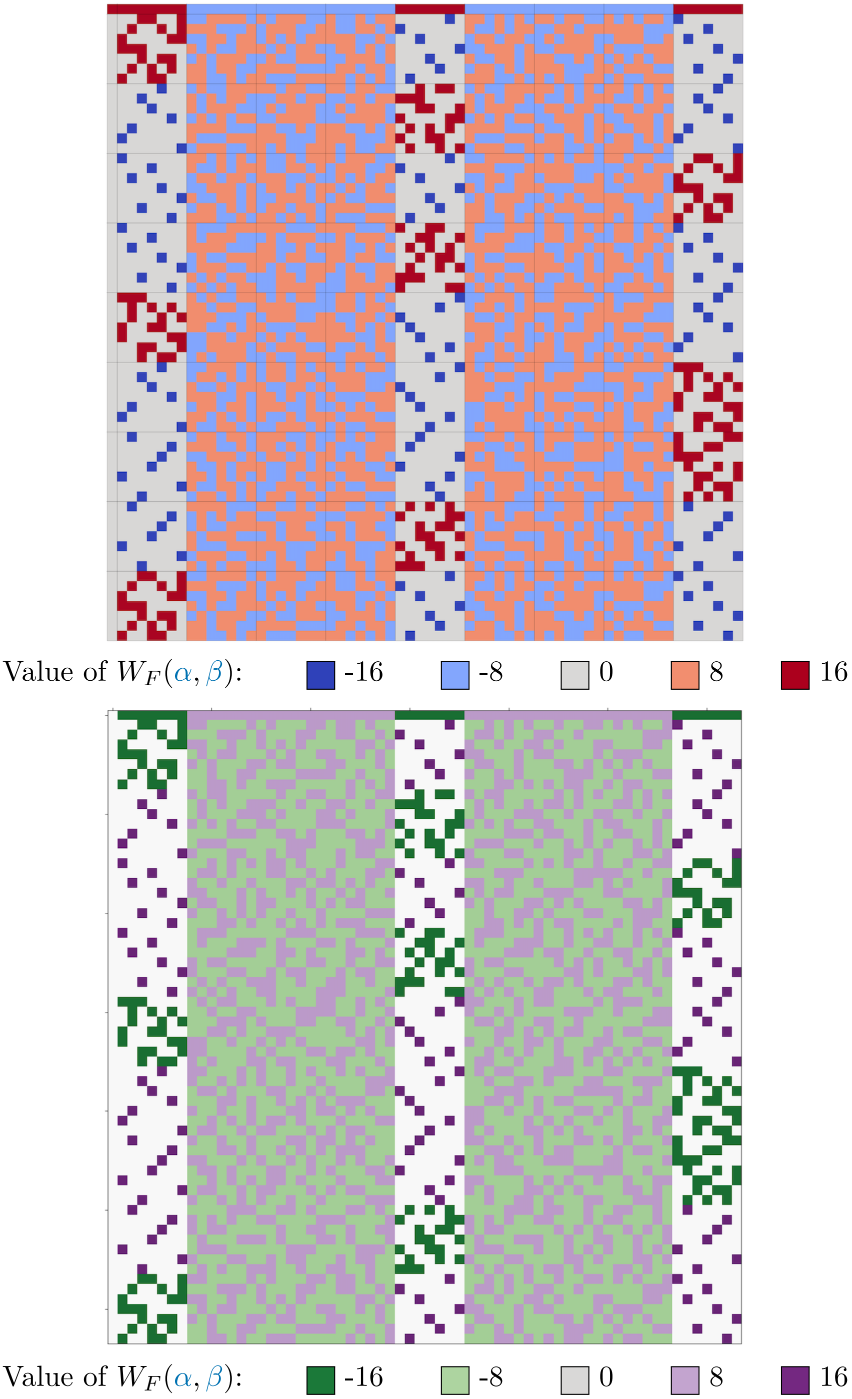
Furthermore, if there exists $x \in \mathbb{F}_{2^k}^*$ such that $\gcd(d_i, \mathrm{ord}(x)) = 1$ for all $i \in [\![1, \ell]\!]$, then the greatest common divisor of the numbers of occurrences of the values in the Walsh spectrum is divisible by $\mathrm{ord}(x)$.

The symmetries highlighted in Corollary 6.52 appear very clearly in the graphical representations of the linear approximation table (LAT) of the Kim mapping and of the Gold power mapping $x \mapsto x^3$ over $\mathbb{F}_{64}$ that are depicted in Figures 6.2 and 6.3. The same property can be stated for the differential distribution table (DDT) of a linearly self-equivalent mapping.



The Walsh coefficients $W_F(\alpha,\beta)$ are enumerated cosets by cosets, β along the x-axis and α along the y-axis.

Figure 6.2: LAT of the Kim mapping using two different colormaps.



The Walsh coefficients $W_F(\alpha, \beta)$ are enumerated cosets by cosets, β along the x-axis and α along the y-axis.

Figure 6.3: LAT of the mapping $x \mapsto x^3$ over $\mathbb{F}_{64}$ using two different colormaps.

Lemma 6.54 (Linear equivalence and DDT)**.** *Let $A, B\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ be bijective linear mappings. Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$. If F satisfies $B \circ F \circ A = F$, then:*

$$\forall \alpha \in \mathbb{F}_2^n, \forall \beta \in \mathbb{F}_2^n, \quad \delta_F(\alpha, \beta) = \delta_F(A(\alpha), B^{-1}(\beta)),$$

where, as defined in Section 2.3.3.b, $\delta_F(\alpha, \beta) = |\{x \in \mathbb{F}_2^n, F(x+\alpha) + F(x) = \beta\}|$.

As in Corollary 6.52, the divisibility of the number of occurrences in the differential spectrum (with non-zero coefficients) also holds for cyclotomic mappings. These properties were already used in a cryptographic context in [Jeo+22]. Indeed, in this paper, the authors use this redundancy among the Walsh and differential spectra to avoid going through all the coefficients while computing the linearity and the differential uniformity of the functions they study.

In our case, these properties can be used as a tool to search for signs of existence of linearly self-equivalent representatives within an equivalence class. Indeed, the Walsh spectrum is preserved by linear equivalence and the differential and extended Walsh spectra are preserved by CCZ-equivalence. However, APN functions all share the same differential spectrum, so this is of low interest. Furthermore, most of the APN functions in the infinite families that we know also share the same Walsh spectrum, as recalled in Section 6.5.1.

6.4.2 Ortho-derivatives of linearly self-equivalent mappings

We then see how to capture linear self-equivalence in another way by using the so-called *ortho-derivative* of quadratic functions.

Definition 6.55 (Ortho-derivative [CCP22, CCP24])**.** Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ be a quadratic function. We say that $\pi\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ is *an* ortho-derivative for F if, for any x and Δ in $\mathbb{F}_2^n$:

$$\pi(\Delta) \cdot (F(x) + F(x+\Delta) + F(0) + F(\Delta)) = 0$$

The set of all ortho-derivatives of F is denoted by $\Pi(F)$. ▷

Note that $\Pi(F)$ is actually a vector space, as the zero function is obviously an ortho-derivative and it is stable by the addition of functions.

As the Walsh transform, an ortho-derivative depends on a specific scalar product. Depending on the domain of the function, we continue using the standard dot product or the ones defined by Eqs. (6.5) and (6.6).

In the case of a quadratic APN function, because the image set of any non-zero derivative is a hyperplane, there exists a single *non-trivial* ortho-derivative, that is $\pi \in \Pi(F)$ such that $\pi(0) = 0$ and $\pi(a) \neq 0$ for any $a \neq 0$. In the following, we refer to this single non-trivial ortho-derivative as *the* ortho-derivative of a quadratic APN function.

The main advantage of the ortho-derivative of a quadratic function is its behaviour within a given EA-equivalence class.

Proposition 6.56 (Ortho-derivative and EA class [CCP22, Proposition 36])**.** *Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ be a quadratic function, $A, B\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ be bijective affine mappings, and $C\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ be an affine function. Let π_F be an ortho-derivative of F. Let G and τ be defined by:*

$$G = B \circ F \circ A + C, \quad \tau = (L_B^*)^{-1} \circ \pi_F \circ L_A,$$

where L_A, L_B are the linear parts of A, B. Then τ is an ortho-derivative of G, that is, $\tau \in \Pi(G)$. In other words, we have:

$$\Pi(G) = (L_B^*)^{-1}\Pi(F)L_A.$$

Proof. Let $\Delta \in \mathbb{F}_2^n$. Then, for any $x \in \mathbb{F}_2^n$, it holds that:

$$\begin{aligned} 0 &= \pi(\Delta) \cdot \big(F(x) + F(x+\Delta) + F(0) + F(\Delta)\big) \\ &= \pi(\Delta) \cdot \big(B^{-1}GA^{-1}(x) + B^{-1}GA^{-1}(x+\Delta) + B^{-1}GA^{-1}(0) + B^{-1}GA^{-1}(\Delta)\big), \end{aligned}$$

because $C(x) + C(x+\Delta) + C(0) + C(\Delta) = 0$ due to the fact that C is affine. Furthermore, because $4B^{-1}(0) = 0$, we obtain:

$$\begin{aligned} 0 &= \pi(\Delta) \cdot \big(L_{B^{-1}}GA^{-1}(x) + L_{B^{-1}}GA^{-1}(x+\Delta) + L_{B^{-1}}GA^{-1}(0) + L_{B^{-1}}GA^{-1}(\Delta)\big) \\ &= (L_{B^{-1}}^* \circ \pi(\Delta)) \cdot \big(GA^{-1}(x) + GA^{-1}(x+\Delta) + GA^{-1}(0) + GA^{-1}(\Delta)\big). \end{aligned}$$

By using $y \leftarrow L_{A^{-1}}(x)$ and $\Delta' \leftarrow L_{A^{-1}}(\Delta)$ as changes of variables, and by introducing $c = A^{-1}(0)$, we obtain, for any y, Δ':

$$\begin{aligned} 0 &= (L_{B^{-1}}^* \circ \pi \circ L_A(\Delta')) \cdot \big(G(y+c) + G(y+\Delta'+c) + G(c) + G(\Delta'+c)\big) \\ &= (L_{B^{-1}}^* \circ \pi \circ L_A(\Delta')) \cdot \big(D_{\Delta'}G(y+c) + D_{\Delta'}G(c)\big) \\ &= (L_{B^{-1}}^* \circ \pi \circ L_A(\Delta')) \cdot \big(D_{\Delta'}G(y) + D_{\Delta'}G(0)\big), \end{aligned}$$

where we use for the last equality the fact that $D_{\Delta'}G$ is affine. This proves that $L_{B^{-1}}^* \circ \pi \circ L_A$ is indeed an ortho-derivative of G. The vector space equality is an immediate consequence of this formula. □

Corollary 6.57. *Let F be a quadratic APN function. Let G be EA-equivalent to F. Then their unique non-zero ortho-derivative are linearly equivalent.*

Proof. This is a direct consequence of the uniqueness of the non-zero ortho-derivative for each function F and G. □

In our case, this also implies the following easy, but important proposition.

Proposition 6.58. *Let F be a quadratic APN function. Let us suppose that F is linearly self-equivalent: $B \circ F \circ A = F$. Let G be EA-equivalent to F: $G = D \circ F \circ E + C$. Then:*

(i) *the ortho-derivative of F is linearly self-equivalent: $(B^{-1})^* \circ \pi_F \circ A = \pi_F$.*

(ii) *the ortho-derivative of G is linearly self-equivalent.*

Proof. By a direct application of Proposition 6.56, we obtain $(B^{-1})^* \circ \pi_F \circ A = \pi_F$, because A, B are linear and, $\pi_G = (L_D^*)^{-1} \circ \pi_F \circ L_E$, *i.e.* $L_D^* \circ \pi_G \circ L_E^{-1} = \pi_F$. By substituting π_F in the formula deduced from self equivalence, we obtain:

$$(B^{-1})^* \circ L_D^* \circ \pi_G \circ L_E^{-1} \circ A = L_D^* \circ \pi_G \circ L_E^{-1},$$

or equivalently:

$$\left((L_D^*)^{-1} \circ (B^{-1})^* \circ L_D^*\right) \circ \pi_G \circ \left(L_E^{-1} \circ A \circ L_E\right) = \pi_G. \tag{6.7}$$

□

In the case of a cyclotomic mapping, or more generally of an ℓ-variate projective mapping, we obtain the following interpretation of the previous proposition.

Corollary 6.59 (Ortho-derivatives of ℓ-variate projective mappings)**.** *Let $F\colon \mathbb{F}_{2^k}^\ell \to \mathbb{F}_{2^k}^\ell$ be a quadratic APN function. Let us suppose that F is an ℓ-variate projective mapping with exponents $(d_1, \dots, d_\ell)$. Then π_F is an ℓ-variate projective mapping with exponents $(-d_1, \dots, -d_\ell)$, where the exponents are considered modulo $2^k - 1$. In particular, the ortho-derivative of a quadratic APN cyclotomic mapping of exponent d is cyclotomic of exponent $-d$, and the ortho-derivative of the power mapping $x \mapsto x^d$ is the power mapping $x \mapsto x^{-d}$.*

Proof. First, we observe that for any $x, y, z \in \mathbb{F}_{2^n}$: $\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_2}(x(yz)) = \mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_2}((yx)z)$, so the multiplication $M_{y,n}$ is its own adjoint, for any $y \in \mathbb{F}_{2^n}$. When looking at functions $F\colon \mathbb{F}_{2^k}^\ell \to \mathbb{F}_{2^k}^\ell$, and using $(x, y) \mapsto \sum_{i=1}^{\ell} \mathrm{Tr}_{\mathbb{F}_{2^k}/\mathbb{F}_2}(x_i y_i)$ as the scalar product, this yields the announced result by using Proposition 6.58. □

In particular, both the linear approximation table (LAT) and the difference distribution table (DDT) of such ortho-derivatives inherit from the symmetries mentioned in Corollary 6.52 and Lemma 6.54. Contrary to the initial quadratic APN functions, their ortho-derivatives are neither quadratic, nor APN. In particular, there is *a priori* no reason for ortho-derivatives to share the same differential spectrum or extended Walsh spectrum. In practice, two functions in two distinct EA-equivalence classes have distinct spectra. This is the reason why these spectra are used as strong invariants of EA-equivalence class for quadratic APN functions, for instance in [CCP22, Table VII], [BIK23, Tables 3 & 4], or [BL22, YP22].

In our case, if the divisibility condition mentioned in Corollary 6.52 does not hold for the ortho-derivative, this proves that the function we consider is not EA-equivalent to a cyclotomic mapping. On the other hand, as there is no known reason for such a structure to randomly occur, this could provide a way to detect the existence of a possible self-equivalent representative.

Example 6.60 (Quadratic APN functions of the Banff list)**.** Among the 13 representatives of quadratic APN functions in 6 variables, as given in the Banff list [Dil09], we can exclude the existence of cyclotomic mappings in the EA-equivalence classes of 9 of them by studying the differential spectra and Walsh spectra of their ortho-derivatives. Indeed, as shown in Table 6.5, in these 9 cases, for at least one of the two spectra, the greatest common divisor of the numbers of occurrences of each value is equal to 1. The four others classes are represented by:

$$\begin{aligned}
P_2 &= X^3, \\
P_3 &= X^3 + a^{11}X^6 + aX^9, \\
P_4 &= a^7X^3 + X^5 + a^3X^9 + g^4X^{10} + X^{17} + a^6X^{18}, \text{ and} \\
P_5 &= X^3 + X^{10} + aX^{24},
\end{aligned}$$

where a is a root of $X^6 + X^4 + X^3 + X + 1$ and where the GCDs corresponding to P_i are given on row $\#i$ of Table 6.5. Among them, P_2 is the cube power mapping, and P_3 is a cyclotomic mapping of exponent 0 with respect to $\mathbb{F}_4$, but the GCD for the Walsh spectrum of its ortho-derivative is 21 and the one for the differential spectrum is 63, which might suggest a property related to $\mathbb{F}_8$ or to the group $\mathbb{G}$ with 21 elements. The representative P_4 is not cyclotomic but the GCDs are in that case equal to 21 and 14, which again suggests a property related to $\mathbb{F}_8$. The polynomial P_5 is the Kim mapping, which is cyclotomic of exponent 3 with respect to $\mathbb{F}_8$. ▷

ID	GCD for Walsh spectrum of F	GCDs for Walsh and differential spectra of the ortho-derivative π_F	Number of mappings
#1	42	(1, 1)	7
#2	42	(84, 63)	1
#3	42	(21, 63)	1
#4	42	(21, 14)	1
#5	42	(28, 21)	1
#6	42	(1, 2)	1
#7	1	(2, 1)	1

Table 6.5: Divisibilities of the numbers of occurrences of each value in the Walsh spectrum of the 6-bit APN functions from the Banff list, and of the Walsh and differential spectra of their ortho-derivatives.

ID	GCD for Walsh spectrum of F	GCDs for Walsh and differential spectra of the ortho-derivative π_F	Number of mappings
#1	4088	(7, 7)	33
#2	4088	(1, 1)	2

Table 6.6: Divisibilities of the numbers of occurrences of each value in the Walsh spectrum of the 9-bit APN functions from [BL22], and of the Walsh and differential spectra of their ortho-derivatives.

ID	GCD for Walsh spectrum of F	GCDs for Walsh and differential spectra of the ortho-derivative π_F	Number of mappings
BL-1	340	(1, 3)	8667
BL-2	2	(1, 3)	3206
BL-3	340	(1, 6)	403
BL-4	4	(1, 3)	311
BL-5	340	(1, 1)	204
BL-6	2	(1, 1)	45
BL-7	340	(1, 12)	26
BL-8	4	(1, 6)	11
BL-9	4	(1, 1)	11
BL-10	340	(1, 15)	10
BL-11	340	(1, 2)	7
BL-12	1	(1, 3)	4
BL-13	340	(1, 24)	3
BL-14	2	(1, 15)	3
BL-15	2	(1, 6)	3
BL-16	340	(1, 5)	2
BL-17	340	(1, 30)	2
BL-18	340	(5, 15)	2
BL-19	340	(2, 2)	1
BL-20	2	(1, 5)	1
BL-21	4	(2, 3)	1
QAM-1	340	(1, 1)	12201
QAM-2	2	(1, 1)	796
QAM-3	340	(1, 2)	359
QAM-4	340	(1, 3)	160
QAM-5	340	(1, 4)	17
QAM-6	2	(1, 3)	14
QAM-7	4	(1, 1)	14
QAM-8	340	(1, 6)	8
QAM-9	340	(1, 5)	8
QAM-10	340	(1, 12)	3
QAM-11	4	(1, 3)	2
QAM-12	340	(1, 10)	2
QAM-13	340	(85, 510)	1
QAM-14	340	(85, 1020)	1
QAM-15	340	(5, 60)	1
QAM-16	340	(2, 2)	1
QAM-17	340	(1, 24)	1
QAM-18	340	(1, 8)	1
QAM-19	2	(1, 2)	1

Table 6.7: Divisibilities of the numbers of occurrences of each value in the Walsh spectrum of the 8-bit APN functions from [BL22] (upper half) and [YP22, YWL14] (lower half), and of the Walsh and differential spectra of their ortho-derivative.

In light of Theorem 6.43, the previous example shows that (most of) the known infinite families of quadratic APN functions have very specific properties. This is also highlighted in Table 6.7 with the 8-bit APN functions found in [BL22, YP22, YWL14]. In particular, while the functions from [BL22] are all linearly self-equivalent, none of them, except maybe the two whose GCDs appear on row BL-18, is EA-equivalent to a non-trivial cyclotomic mapping. On the other hand, we observe in Table 6.6 that out of the 35 known 9-bit quadratic APN functions from [BL22], 33 could potentially be EA-equivalent to cyclotomic mappings with respect to $\mathbb{F}_8$.

6.4.3 Searching for linearly self-equivalent mappings within an EA- or CCZ-equivalence class

We now discuss how we could determine in the general (non-quadratic) case whether a function is EA-equivalent or CCZ-equivalent to a cyclotomic mapping or to an ℓ-variate projective mapping.

The following approach is in line with the proof of Proposition 6.58. Let F be linearly self-equivalent: $B \circ F \circ A = F$. Let G be EA-equivalent to F such that they satisfy $F = D \circ G \circ E + C$. Then it holds that:

$$B \circ (D \circ G \circ E + C) \circ A = D \circ G \circ E + C,$$

or equivalently:

$$B \circ D \circ G \circ E \circ A = D \circ G \circ E + C + B \circ C \circ A.$$

By composing the output by D^{-1} and the input by E^{-1}, this is equivalent to:

$$(D^{-1} \circ B \circ D) \circ G \circ (E \circ A \circ E^{-1}) = G + D^{-1}(C + B \circ C \circ A) \circ E^{-1}.$$

In other words, G is EA self-equivalent. Furthermore $D^{-1} \circ B \circ D$ is an affine mapping with $L_D^{-1} \circ B \circ L_D$ as linear part, and a similar property holds for $E \circ A \circ E^{-1}$. This implies that the minimal polynomials of the involved transformations are preserved by EA-equivalence. This can again give proofs of the non-existence of cyclotomic (or ℓ-variate projective) representatives within an EA-equivalence class.

Actually, the same technique can be adapted to the case of CCZ-equivalence.

Proposition 6.61 (CCZ-equivalent functions to a linearly self-equivalent one)**.** *Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ be a linearly self-equivalent mapping satisfying $B \circ F \circ A = F$ for some linear bijections A, B. Let G be CCZ-equivalent to F. Then:*

(i) *G is CCZ self-equivalent.*

(ii) *There exists an $\mathbb{F}_2$-affine bijective mapping $\mathcal{A}\colon (\mathbb{F}_2^n)^2 \to (\mathbb{F}_2^n)^2$ with linear part $\mathcal{L}$ such that $\mathcal{A}(\mathcal{G}_G) = \mathcal{G}_G$, and $\mathcal{L}$ is similar to $\mathrm{diag}(A, B)$. Most notably, $\mathrm{diag}(A, B)$ and $\mathcal{L}$ have the same canonical form and $\min(\mathcal{L}) = \mathrm{lcm}(\min(A), \min(B))$.*

Proof. From the two hypotheses, it holds that:

$$\begin{pmatrix} A & 0 \\ 0 & B \end{pmatrix} \mathcal{G}_F = \mathcal{G}_F, \quad \text{and } \mathcal{A}(\mathcal{G}_G) = \mathcal{G}_F,$$

for some affine bijection $\mathcal{A}\colon (\mathbb{F}_2^n)^2 \to (\mathbb{F}_2^n)^2$. By substituting $\mathcal{G}_F$ by $\mathcal{A}(\mathcal{G}_G)$ in the first equality, we obtain:

$$\begin{pmatrix} A & 0 \\ 0 & B \end{pmatrix} \circ \mathcal{A}(\mathcal{G}_G) = \mathcal{A}(\mathcal{G}_G) \quad \iff \quad \mathcal{A}^{-1} \circ \begin{pmatrix} A & 0 \\ 0 & B \end{pmatrix} \circ \mathcal{A}(\mathcal{G}_G) = \mathcal{G}_G.$$

In other words, G is CCZ self-equivalent. Furthermore, the linear part of the affine mapping $\mathcal{A}^{-1} \circ \mathrm{diag}(A, B) \circ \mathcal{A}$ is $\mathcal{L}^{-1} \circ \mathrm{diag}(A, B) \circ \mathcal{L}$, which is similar to $\mathrm{diag}(A, B)$. In particular, $\mathcal{L}$ and $\mathrm{diag}(A, B)$ have the same minimal polynomial, which is the least common multiple of the ones of A and B. □

Contrary to ortho-derivatives which are not defined for functions of degree more than 2, this method can be applied to any function.

Example 6.62 (Brinckmann-Leander-Edel-Pott APN cubic [BL08, EP09])**.** Let us consider the well-known Brinckmann-Leander-Edel-Pott APN cubic for $n = 6$ [BL08, EP09]. Then there is neither cyclotomic nor ℓ-variate projective mapping in its CCZ-equivalence class. Indeed, its 7 non-trivial automorphisms share the same elementary divisors, which are $X+1$ with multiplicity 2 and $(X+1)^2$ with multiplicity 5. In particular, if a linearly self-equivalent function exists in its CCZ-equivalence class, then, according to Proposition 6.61, the canonical form of $\mathrm{diag}(A, B)$ is the same as the one of $\mathcal{L}$. But the canonical form of $\mathrm{diag}(A, B)$ is the concatenation of the ones of A and B. This implies that the canonical form of A is made of blocks $C(X+1)$ or $C\left((X+1)^2\right)$ with at least one block $C\left((X+1)^2\right)$. Therefore, because of Proposition 6.30, $\min(A) = (X+1)^2$. Thus $\min(A)$ is not irreducible which contradicts the hypotheses of Theorem 6.33 for the cyclotomic case, and the ones of Theorem 6.40 for the ℓ-variate projective case. Indeed, the only non-trivial subfields of $\mathbb{F}_{2^6}$ are $\mathbb{F}_{2^2}$ and $\mathbb{F}_{2^3}$, with both $2^2 - 1$ and $2^3 - 1$ being prime. ▷

In particular, this example generalizes the well-known fact that this function is not CCZ-equivalent to a monomial mapping. Actually, these non-LE automorphisms correspond to the 7 affine derivatives of the Brinckmann-Leander-Edel-Pott cubic. By definition, any such automorphism corresponds to a triangular block matrix with a diagonal made of identity blocks:

$$\begin{pmatrix} \mathrm{Id} & 0 \\ L & \mathrm{Id} \end{pmatrix} + \begin{pmatrix} \Delta^{\mathrm{in}} \\ \Delta^{\mathrm{out}} \end{pmatrix} \in \mathrm{Aut}(F),$$

where L is the linear part of the derivative $D_{\Delta^{\mathrm{in}}} F$ and Δ^{out} its constant term. In particular, the linear part of such an EA automorphism is involutive: its canonical form is therefore only made of blocks $C(X+1)$ and $C((X+1)^2)$. The argument used in the previous example can therefore be generalized. Indeed, from an EA automorphism related to an affine derivative of a function F, we can never prove the existence of a linearly self-equivalent function G in its CCZ-equivalence class, where G satisfies $B \circ G \circ A = G$, with *non-involutive* A and/or B. Conversely, if the only non-trivial automorphisms of a function come from its affine derivatives, neither cyclotomic nor ℓ-variate cyclotomic mapping exists in its CCZ-equivalence class. This also implies the non-existence of representatives that commute with the Frobenius automorphism because $x \mapsto x^2$ is not involutive.

A lot of questions still remain open. For instance, this does not rule out the existence of a linearly self-equivalent mapping G in the CCZ-equivalence class of this cubic, it only proves that if such a mapping G exists, it satisfies $B \circ G \circ A = G$ for two involutions A and B. However, in light of Theorem 6.43, it proves that this cubic is very different from the other known APN functions. The most interesting problem that remains is the following one.

Problem 6.63 (From CCZ self-equivalence to linear self-equivalence)**.** *Given a CCZ self-equivalent function, is it possible to use its automorphisms to find a linearly self-equivalent function in the same CCZ-equivalence class ?*

6.5 (Quadratic) APN mappings

This section is dedicated to APN mappings, and in particular to quadratic APN mappings. First, we recall the main results about the Walsh spectrum analysis of most quadratic APN functions. As explained below, their Walsh spectrum is very structured in most cases. In particular, in even dimension, a lot of cases coincide with the Walsh spectrum shared by very specific cyclotomic mappings. This also explains why the symmetries in the Walsh spectrum should be studied on the ortho-derivative, rather than on the function itself.

6.5.1 Classical Walsh spectra

6.5.1.a The case of odd dimension

First, we recall some definitions related to quadratic APN functions.

Definition 6.64 (Plateaued function [ZZ99])**.** Let $f\colon \mathbb{F}_2^n \to \mathbb{F}_2$. The function f is said to be *plateaued* if there exists a positive integer $c > 0$ such that for any $\alpha \in \mathbb{F}_2^n$, $W_f(\alpha) \in \{0, \pm c\}$. In that case, $c = 2^i$, with $i \geq \frac{n}{2}$, and the numbers of occurrences of $0, 2^i, -2^i$ in the Walsh spectrum are given in Table 6.8. The number $c = 2^i$ is called the *amplitude* of f.

A vectorial function $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ is called plateaued (resp. plateaued with single amplitude) if all its non-zero components are plateaued (resp. plateaued with the same amplitude). ▷

Value	Multiplicity
2^i	$2^{2n-2i-1} + (-1)^{f(0)} 2^{n-i-1}$
0	$2^n - 2^{2n-2i}$
-2^i	$2^{2n-2i-1} - (-1)^{f(0)} 2^{n-i-1}$

Table 6.8: Walsh spectrum of a plateaued Boolean function of n variables with amplitude 2^i.

Example 6.65 (Quadratic functions)**.** Any quadratic function $f\colon \mathbb{F}_2^n \to \mathbb{F}_2$ is actually plateaued. Indeed, for any $\alpha \in \mathbb{F}_2^n$:

$$W_f(\alpha)^2 = \sum_{\Delta \in \mathbb{F}_2^n} (-1)^{\alpha \cdot \Delta} W_{D_\Delta f}(0).$$

This is for instance proven in Eq. (2.20). Because f is quadratic, its derivatives are either affine or constant, so we further get:

$$W_f(\alpha)^2 = 2^n \sum_{\Delta \in \mathbb{F}_2^n} (-1)^{\alpha \cdot \Delta + D_\Delta f(0)} \mathbf{1}_{\mathrm{LS}(f)}(\Delta) = 2^n \sum_{\Delta \in \mathrm{LS}(f)} (-1)^{\alpha \cdot \Delta + D_\Delta f(0)}.$$

But the function $\Delta \mapsto D_\Delta f(0)$ is linear over $\mathrm{LS}\,(f)$ with corresponding mask ε (because f is affine over $\mathrm{LS}\,(f)$, see Lemma 4.20), so:

$$W_f(\alpha)^2 = 2^n \sum_{\Delta \in \mathrm{LS}(f)} (-1)^{(\alpha+\varepsilon)\cdot \Delta} = 2^{n+\dim(\mathrm{LS}(f))} \mathbf{1}_{\mathrm{LS}(f)^\perp}(\alpha + \varepsilon).$$

This proves that f is plateaued with amplitude $2^{\frac{n+\dim(\mathrm{LS}(f))}{2}}$. ▷

Definition 6.66 (Almost bent (AB) function [CV95])**.** Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$. Then $\mathcal{L}(F) \geq 2^{\frac{n+1}{2}}$. If F satisfies $\mathcal{L}(F) = 2^{\frac{n+1}{2}}$ then F is said to be *almost bent* (AB). In that case n is necessarily odd, and for any $\alpha \in \mathbb{F}_2^n, \beta \in \mathbb{F}_2^n \setminus \{0\}$:

$$W_F(\alpha, \beta) \in \left\{0, \pm 2^{\frac{n+1}{2}}\right\},$$

that is F, is plateaued with single amplitude $2^{\frac{n+1}{2}}$. ▷

The previously mentioned definition and properties are due to Chabaud & Vaudenay [CV95]. In our case, we are interested in the following relations between the APN and almost bent properties.

Proposition 6.67 (APN and AB functions, odd case)**.** *Let n be odd. Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$. Then F is almost bent if and only if F is APN and F is plateaued.*

The direct implication appears in [CV95, Theorem 4 & Note 2]. The indirect one is proven in the quadratic case in [CCZ98, Theorem 8], and the general case in [CCD00, Corollary 4.2].

Any quadratic APN function in odd dimension is therefore plateaued with single amplitude $2^{\frac{n+1}{2}}$. This implies that its Walsh spectrum can be computed from Table 6.8, by also taking care of the zero function. We detail it in Table 6.9.

Value	Multiplicity
2^n	1
$2^{\frac{n+1}{2}}$	$(2^n-1)\left(2^{n-2}+2^{\frac{n-3}{2}}\right)$
0	$(2^n-1)(2^{n-1}+1)$
$-2^{\frac{n+1}{2}}$	$(2^n-1)\left(2^{n-2}-2^{\frac{n-3}{2}}\right)$

Table 6.9: Walsh spectrum of an almost bent function over $\mathbb{F}_2^n$ with $F(0)=0$.

Value	Multiplicity
2^n	1
$2^{\frac{n}{2}+1}$	$\frac{1}{3}(2^n-1)(2^{n-3}+2^{\frac{n-4}{2}})$
$2^{\frac{n}{2}}$	$\frac{2}{3}(2^n-1)(2^{n-1}+2^{\frac{n-2}{2}})$
0	$(2^n-1)(2^{n-2}+1)$
$-2^{\frac{n}{2}}$	$\frac{2}{3}(2^n-1)(2^{n-1}-2^{\frac{n-2}{2}})$
$-2^{\frac{n}{2}+1}$	$\frac{1}{3}(2^n-1)(2^{n-3}-2^{\frac{n-4}{2}})$

Table 6.10: Walsh spectrum of an APN function over $\mathbb{F}_2^n$, n even with $\frac{2}{3}(2^n-1)$ bent components and $F(0)=0$.

6.5.1.b The case of even dimension

In even dimension, since almost bent functions do not exist, the situation is a bit different. This is detailed in the following proposition.

Proposition 6.68 (Plateaued APN functions, even case [Ber+06, Corollary 3])**.** *Let n be even. Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ be a plateaued function. Let B be its number of bent components. Then:*

1. *If F is APN, then $\frac{2}{3}(2^n-1) \le B$.*
2. *If F is APN, then $B=\frac{2}{3}(2^n-1)$ if and only if $\mathcal{L}(F)=2^{\frac{n+2}{2}}$.*
3. *If $B=\frac{2}{3}(2^n-1)$ and $\mathcal{L}(F)=2^{\frac{n+2}{2}}$, then F is APN, and it has the Walsh spectrum described in Table 6.10.*

This extremal case with exactly $\frac{2}{3}(2^n-1)$ bent components occurs for almost all known APN functions in even dimension. As an example, this is the case for the Kim mapping and the cube power mapping over $\mathbb{F}_{64}$. The one third/two thirds partition of the non-zero components is highlighted in Figures 6.2 and 6.3. Because the infinite families are all quadratic, this can be proven by studying another extremal behavior, with respect to the size of the image set. This is detailed in the following proposition and corollary.

Proposition 6.69 (Size of the image set [CHP17]). *Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ be a function with differential uniformity δ_F. Then:*

$$\left\lceil \frac{\frac{2^{2n}}{|\mathrm{Im}(F)|} - 2^n}{2^n - 1} \right\rceil \leq \delta_F.$$

Proof.

$$\begin{aligned}
\sum_{a\in\mathbb{F}_2^n, a\neq 0} \left|D_aF^{-1}(\{0\})\right| &= \sum_{a\in\mathbb{F}_2^n, a\neq 0} |\{x \in \mathbb{F}_2^n,\ F(x+a) = F(x)\}| \\
&= \sum_{a\in\mathbb{F}_2^n, a\neq 0} |\{(x, x+a),\ x \in \mathbb{F}_2^n,\ F(x+a) = F(x)\}| \\
&= |\{(x,y) \in \mathbb{F}_2^n \times \mathbb{F}_2^n,\ y \neq x, F(y) = F(x)\}| \\
&= |\{(x,y) \in \mathbb{F}_2^n \times \mathbb{F}_2^n,\ F(y) = F(x)\}| - 2^n \\
&= \sum_{x\in\mathbb{F}_2^n} \left|F^{-1}(\{F(x)\})\right| - 2^n \\
&= \sum_{b\in\mathrm{Im}(F)} \left|F^{-1}(\{b\})\right|^2 - 2^n.
\end{aligned}$$

Therefore, by applying Cauchy-Schwarz inequality to the sequences $(1)_{b\in\mathrm{Im}(F)}$ and $(|F^{-1}(b)|)_{b\in\mathrm{Im}(F)}$we observe that:

$$\frac{2^{2n}}{|\mathrm{Im}(F)|} - 2^n = \frac{(\sum_{b\in\mathrm{Im}(F)} |F^{-1}(b)|)^2}{|\mathrm{Im}(F)|} - 2^n \leq \sum_{b\in\mathrm{Im}(F)} \left|F^{-1}(b)\right|^2 - 2^n = \sum_{a\in\mathbb{F}_2^n, a\neq 0} \left|D_aF^{-1}(0)\right|.$$

This finally implies that there exists $a \neq 0$ such that:

$$\frac{\frac{2^{2n}}{|\mathrm{Im}(F)|} - 2^n}{2^n - 1} \leq \left|D_aF^{-1}(0)\right|;$$

otherwise, the previous inequality would not hold. The announced bound on the differential uniformity is thus obtained. □

Corollary 6.70 (Size of the image set of APN functions [CHP17, KKK23, Cze20]). *Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ be an APN function. Then:*

$$|\mathrm{Im}(F)| \geq \begin{cases} \frac{2^n+1}{3} & \text{if } n \text{ is odd,} \\ \frac{2^n+2}{3} & \text{if } n \text{ is even.} \end{cases} \tag{6.8}$$

Proof. Adapted from [Car21]. $\frac{\frac{2^{2n}}{|\mathrm{Im}(F)|} - 2^n}{2^n - 1} \leq \delta_F$ becomes $\frac{2^{2n}}{3\cdot 2^n - 2} \leq |\mathrm{Im}(F)|$ when $\delta_F = 2$. We further see that:

$$0 < \left|\frac{2^n+2}{3} - \frac{2^{2n}}{3\cdot 2^n - 2}\right| = \frac{4(2^n-1)}{9\cdot 2^n - 6} < 1.$$

so there is at most a single integer in that range, which in that case would be $\left\lceil \frac{2^{2n}}{3\cdot 2^n - 2} \right\rceil$. For an even n, we know that $3 \mid 2^n + 2$, so $\frac{2^n+2}{3} \in \mathbb{N}$. In the same way, for an odd n, $\frac{2^n+1}{3} \in \mathbb{N}$ and lies in $]\frac{2^{2n}}{3\cdot 2^n - 2}, \frac{2^n+2}{3}[$. □

This result was independently obtained in multiple papers [CHP17, KKK23, Cze20]. Note that $\frac{2^n+2}{3} = \frac{2^n-1}{3} + 1$. In particular, in many cases, APN functions reaching this bound are almost 3-to-1 functions (with $F^{-1}(\{0\}) = \{0\}$), see Definition 6.12. It is believed [KKK23] that they are the only APN functions in even dimension that reach the bound of Corollary 6.70.

Example 6.71 (APN monomials)**.** As shown in [KKK23, Corollary 4], Corollary 6.70 gives a straight-forward proof of the fact that an APN function $F\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ defined by $F(x) = x^d$ satisfies $\gcd(d, 2^n-1) = 1$ if n is odd and $\gcd(d, 2^n-1) = 3$ if n is even. Indeed, its image set is the group of the d-th roots of unity which is of cardinality $\frac{2^n-1}{\gcd(d,2^n-1)}$. This implies that $\gcd(d, 2^n-1) \leq 3$, and this common divisor must be odd because 2^n-1 is odd. If n is odd, $3 \nmid 2^n-1$, so necessarily $\gcd(d, 2^n-1) = 1$. When n is even, $\gcd(d, 2^n-1) = 1$ would imply that $x \mapsto x^d$ is bijective over $\mathbb{F}_4 \subset \mathbb{F}_{2^k}$ but there does not exist any APN bijection over $\mathbb{F}_4$, so $\gcd(d, 2^n-1) = 3$. ▷

The following proposition describes in more detail the relation with the almost 3-to-1 property.

Proposition 6.72 (Almost 3-to-1, plateaued & APN functions)**.** *Let n be even and $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$, with $F(0) = 0$. Then:*

1. *[KKK23, Theorem 15] If F is a plateaued almost 3-to-1 function, then F is APN and its Walsh spectrum is the one given in Table 6.10.*

2. *[KKK23, Theorem 4] If F is an APN cyclotomic mapping of exponent 0 with respect to $\mathbb{F}_4$, then F is almost-3-to-1.*

3. *[KKK23, Theorem 5] Furthermore, let F be a plateaued cyclotomic mapping of exponent 0 with respect to $\mathbb{F}_4$. Then the following statements are equivalent:*

   (i) *F is APN.*

   (ii) *F is almost 3-to-1.*

   (iii) *F has the Walsh spectrum given in Table 6.10.*

Proof. We prove Item 3, while Items 1 & 2 can be seen as subcases of this proof. Let us assume that F is a quadratic cyclotomic mapping of exponent 0 with respect to $\mathbb{F}_4$, that is, a 3-divisible function.

(i) $\Longrightarrow$ **(ii)** If F is APN, because of Corollary 6.70, it follows that F is 3-to-1 on $\mathbb{F}_{2^n}^*$.

(ii) $\Longrightarrow$ **(iii)** First, we observe that:

$$\sum_{\beta \in \mathbb{F}_{2^n}} W_F(0, \beta) = \sum_{x \in \mathbb{F}_{2^n}} \sum_{\beta \in \mathbb{F}_{2^n}} (-1)^{\mathrm{Tr}(\beta F(x))} = 2^n |F^{-1}(0)| = 2^n \ ,$$

because 0 has a single preimage. In other words, $\sum_{\beta \in \mathbb{F}_{2^n}^*} W_F(0,\beta) = 0$. Moreover:

$$\begin{aligned}
\sum_{\beta \in \mathbb{F}_{2^n}} W_F(0,\beta)^2 &= \sum_{x \in \mathbb{F}_{2^n}} \sum_{y \in \mathbb{F}_{2^n}} \sum_{\beta \in \mathbb{F}_{2^n}} (-1)^{\mathrm{Tr}(\beta F(x)) + \mathrm{Tr}(\beta F(y))} \\
&= \sum_{x \in \mathbb{F}_{2^n}} \sum_{a \in \mathbb{F}_{2^n}} \sum_{\beta \in \mathbb{F}_{2^n}} (-1)^{\mathrm{Tr}(\beta (F(x+a) + F(x)))} \\
&= 2^n \sum_{x \in \mathbb{F}_{2^n}} |\{a \in \mathbb{F}_{2^n} : F(x+a) + F(x) = 0\}| \\
&= 2^n \left(1 + 3(2^n - 1)\right) = 2^{2n+1} + 2^{2n} - 2^{n+1} \ ,
\end{aligned}$$

where we successively use $y \leftarrow x + a$ as a change of variables, the mean of characters and our hypothesis, that is, that F is almost 3-to-1. It follows that:

$$\sum_{\beta \in \mathbb{F}_{2^n}^*} W_F(0,\beta)^2 = 2^{n+1}(2^n - 1) \ .$$

Therefore:

$$\mathcal{S} \ := \sum_{\beta \in \mathbb{F}_{2^n}^*} \left(W_F(0,\beta)^2 + (-2)^{n/2} W_F(0,\beta) - 2^{n+1} \right) = 0.$$

But $\mathcal{S}$ can also be expressed as:

$$S = \sum_{\beta \in \mathbb{F}_{2^n}^*} \left(W_F(0,\beta) - (-2)^{n/2} \right) \left(W_F(0,\beta) - (-2)^{n/2+1} \right) = 0.$$

Because F is plateaued, all $W_F(0,\beta)$ are divisible by $2^{n/2}$. Moreover, it holds that:

$$W_F(0,\beta) - 1 = 2^n - 2\,|\{x \in \mathbb{F}_{2^n}, \mathrm{Tr}(\beta F(x)) = 1\}| - 1,$$

but $2^n - 1 \equiv 0 \bmod 3$ because n is even and $|\{x \in \mathbb{F}_{2^n}, \mathrm{Tr}(\beta F(x)) = 1\}| \equiv 0 \bmod 3$ because F is almost 3-to-1 and with $F^{-1}(\{0\}) = 0$, so we observe that:

$$W_F(0,\beta) - 1 \equiv 0 \bmod 3,$$

and therefore $W_F(0,\beta) \notin \{0, -(-2)^{n/2}, -(-2)^{n/2+1}\}$, because $-2 \equiv 1 \bmod 3$. When $n/2$ is even, it then holds that:

$$\left(W_F(0,\beta) - (-2)^{n/2} \right) \left(W_F(0,\beta) - (-2)^{n/2+1} \right) \geq 0, \tag{6.9}$$

that is, the two terms are of the same sign. Indeed, if $W_F(0,\beta) - 2^{n/2} \geq 0$, then of course $W_F(0,\beta) \geq -2^{n/2+1}$. Conversely, if $W_F(0,\beta) < 2^{n/2}$, then necessarily $W_F(0,\beta) \leq -2^{n/2+1}$, because $W_F(0,\beta)$ must be divisible by $2^{n/2}$, but cannot belong to $\{0, -2^{n/2}\}$. When $n/2$ is odd, Eq. (6.9) also holds for the same reasons. Indeed, if $W_F(0,\beta) - 2^{n/2+1} \geq 0$, then of course $W_F(0,\beta) \geq -2^{n/2}$. Conversely, if $W_F(0,\beta) < 2^{n/2+1}$, then necessarily

$W_F(0,\beta) \leq -2^{n/2}$, because $W_F(0,\beta)$ must be divisible by $2^{n/2}$, but cannot belong to $\{0, 2^{n/2}\}$.

In any case, we deduce that, because S is a sum of positive terms which vanishes, all its terms must vanish. In other words, $W_F(0,\beta) \in \{(-2)^{n/2}, (-2)^{n/2+1}\}$ and these two values occur with multiplicities $2(2^n-1)/3$ and $(2^n-1)/3$ respectively, because their multiplicities (a,b) are the solutions of the following system:

$$a + b = 2^n - 1, \quad a(-2)^{n/2} + b(-2)^{n/2+1} = 0.$$

Since F is plateaued, the values of β such that $|W_F(0,\beta)| = 2^{n/2}$ correspond to bent components, and the other ones to components with linearity $2^{n/2+1}$, so F has the announced Walsh spectrum.

(ii) $\Longrightarrow$ **(iii)** If F has the Walsh spectrum given in Table 6.10, then F satisfies the hypotheses of Proposition 6.68, Item 3, so F is APN.

□

Some of these results already appear in [CGT16, Corollary 1 & Theorem 3], in the context of quadratic functions.

Remark 6.73. As we will see below, if a cyclotomic mapping with respect to $\mathbb{F}_{2^k}$, with k even, is APN, then necessarily its exponent d defines an APN function $x \mapsto x^d$ over $\mathbb{F}_{2^k}$. Then, as shown in Example 6.71, 3 divides d, but 3 also divides $2^k - 1$ because k is even. This implies that an APN cyclotomic mapping with respect to a subfield of even dimension is necessarily 3-divisible. In particular, it is also almost 3-to-1 because of Corollary 6.70, be it quadratic (or plateaued) or not. However, this does not imply that F has the *classical Walsh spectrum* given in Table 6.10. An example of this situation is Dobbertin's power function over $\mathbb{F}_{2^{10m}}$, $m \geq 1$, with exponent $d = 2^{8m} + 2^{6m} + 2^{4m} + 2^{2m} - 1$ [Dob01]. As any power function, this function is a cyclotomic mapping of exponent 3 with respect to $\mathbb{F}_{2^{2m}}$: it can be written as $x^3 P(x^{2^{2m}-1})$. It is also 3-to-1 by the previous discussion, however it is known that its Walsh spectrum is not $2^{n/2}$ divisible [CCD00]. Then, its Walsh spectrum cannot be the classical one given in Table 6.10, and its form is only conjectured [Bud+22, Conjecture 29]. ▷

6.5.2 APN (generalized) cyclotomic mappings

We have shown in Section 6.3 that many families among the known APN functions are linearly equivalent to a cyclotomic mapping. We then further study these mappings since they seem to play a particular role among all known APN mappings. Note that the differential uniformity of cyclotomic mappings was investigated by Chen and Coulter [CC23] recently, but their results do not provide any relevant information for our parameters in characteristic 2. We thus continue studying the link between those two properties. Since the generalization is straightforward, we state the results in a more general context, such as the one of generalized cyclotomic mappings. However, as seen in Section 6.1.6, they do not *a priori* provide linear self-equivalence, so the main concern remains the specific case of cyclotomic mappings.

6.5.2.a Necessary conditions to be APN

First of all, an APN generalized cyclotomic mapping with respect to a subfield $\mathbb{F}_{2^k}$ has a single preimage for 0.

Proposition 6.74. *Let $F\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ be a generalized cyclotomic mapping with respect to $\mathbb{F}_{2^k}$. Assume that there exists $\lambda \neq 0$ such that $F(\lambda) = 0$. Then the differential uniformity of F is at least 2^k.*

Proof. We observe that, for any $\varphi \in \mathbb{F}_{2^k}$, we have $F(\lambda\varphi + \lambda) + F(\lambda\varphi) = 0$. □

In particular an APN generalized cyclotomic mappping with respect to $\mathbb{F}_{2^k}$ must satisfy $F^{-1}(\{0\}) = \{0\}$. Furthermore, it must be based on APN monomials over $\mathbb{F}_{2^k}$.

Lemma 6.75. *Let $F : \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ be a generalized cyclotomic mapping with respect to a subfield $\mathbb{F}_{2^k}$. If F is APN, then all its exponents d_λ, $\lambda \neq 0$, defined in Definition 6.25 are such that $x \mapsto x^{d_\lambda}$ is APN on $\mathbb{F}_{2^k}$.*

Proof. Suppose that there exists a coset $\lambda\mathbb{F}_{2^k}$, $\lambda \neq 0$, such that $G_\lambda : x \mapsto x^{d_\lambda}$ is not APN on $\mathbb{F}_{2^k}$. Then, there exist $\varphi_1, \varphi_2 \in \mathbb{F}_{2^k}$ such that $\delta_{G_\lambda}(\varphi_1, \varphi_2) > 2$. Using that, for any $\varphi \in \mathbb{F}_{2^k}$,

$$F(\lambda\varphi + \lambda\varphi_1) + F(\lambda\varphi) = F(\lambda)\left(G_\lambda(\varphi + \varphi_1) + G_\lambda(\varphi)\right),$$

we deduce that there exist more than two $\varphi \in \mathbb{F}_{2^k}$ such that

$$F(\lambda\varphi + \lambda\varphi_1) + F(\lambda\varphi) = F(\lambda)\varphi_2$$

implying that F is not APN. □

This explains the fact that the exponents of the cyclotomic mappings in most of the infinite families are Gold exponents.

Note that the stability of $\mathbb{F}_{2^k}$ with respect to addition is used in the previous proof, this is the reason why it cannot (at least directly) be adapted to the more general case of cyclotomic mappings with respect to generic groups $\mathbb{G}$.

The case of subfields of even dimension is very peculiar. Indeed, we can in that case derive the following necessary conditions from Corollary 6.70, some of which are already mentioned in Remark 6.73.

Proposition 6.76. *Let k be an even divisor of n and F be a cyclotomic mapping with respect to $\mathbb{F}_{2^k}$. Let Γ be a system of representatives of the multiplicative cosets of $\mathbb{F}_{2^k}$. If F is APN then:*

- *F does not satisfy the $\mathbb{F}_{2^k}$-subspace property,*

- *F is cyclotomic of exponent 0 with respect to $\mathbb{F}_4$,*
- *all $F(\gamma)$, $\gamma \in \Gamma$ belong to different cosets of $\mathbb{F}_{2^k}$,*
- *F is almost 3-to-1 with $F^{-1}(0) = \{0\}$.*

Proof. Lemma 6.75 states that x^d is APN over $\mathbb{F}_{2^k}$. The first point is then a consequence of the property exhibited in Example 6.71, because there does not exist any APN bijective monomial for even dimension. Furthermore, because k is even, we have that $\gcd(d, 2^k - 1) = 3$, which implies that F is 3-divisible by Proposition 6.13. From Corollary 6.70, F is then necessarily almost 3-to-1, or equivalently, all $F(\gamma), \gamma \in \Gamma$ belong to different cosets of $\mathbb{F}_{2^k}$. □

6.5.2.b Spectral properties of (generalized) cyclotomic mappings

For generalized cyclotomic mappings with respect to a subfield, we can easily express the Walsh coefficients in terms of the Walsh coefficients of the power functions $x \mapsto x^{d_\lambda}$.

Proposition 6.77. *Let $n = \ell k$ and $F\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ be a generalized cyclotomic mapping with respect to $\mathbb{F}_{2^k}$. Let $d_\gamma, \gamma \in \Gamma$ denote its exponents as defined in Definition 6.25, where Γ is a system of representatives of the multiplicative cosets of $\mathbb{F}_{2^k}$. For any $\alpha, \beta \in \mathbb{F}_{2^n}$, we have:*

$$W_F(\alpha, \beta) = -\sum_{i=1}^{\ell-1} 2^{ik} + \sum_{\gamma\in\Gamma} W_{\mathbb{F}_{2^k}, x^{d_\gamma}} \left(\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\alpha\gamma), \mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\beta F(\gamma))\right).$$

Proof. A direct computation yields:

$$\begin{aligned}
W_F(\alpha, \beta) &= \sum_{\lambda\in\mathbb{F}_{2^n}} (-1)^{\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\alpha\lambda+\beta F(\lambda))} \\
&= (-1)^0 + \sum_{\gamma\in\Gamma} \sum_{\varphi\in\mathbb{F}_{2^k}^*} (-1)^{\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\alpha\gamma\varphi+\beta F(\gamma\varphi))} \\
&= 1 + \sum_{\gamma\in\Gamma} \left(\sum_{\varphi\in\mathbb{F}_{2^k}} (-1)^{\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\alpha\gamma\varphi+\beta F(\gamma\varphi))} - 1 \right) \\
&= (1 - |\Gamma|) + \sum_{\gamma\in\Gamma} \sum_{\varphi\in\mathbb{F}_{2^k}} (-1)^{\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\alpha\gamma\varphi+\beta F(\gamma\varphi))} \\
&= -\sum_{i=1}^{\ell-1} 2^{ik} + \sum_{\gamma\in\Gamma} \sum_{\varphi\in\mathbb{F}_{2^k}} (-1)^{\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\alpha\gamma\varphi+\beta F(\gamma)\varphi^{d_\gamma})} \\
&= -\sum_{i=1}^{\ell-1} 2^{ik} + \sum_{\gamma\in\Gamma} W_{\mathbb{F}_{2^k}, x^{d_\gamma}} \left(\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\alpha\gamma), \mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\beta F(\gamma))\right);
\end{aligned}$$

where we successively used the multiplicative decomposition using Γ, changed the sum over $\mathbb{F}_{2^k}^*$ into a sum over $\mathbb{F}_{2^k}$, used Definition 6.25, and finally the trace linearity. □

Proposition 6.78 (Walsh coefficients in zero)**.** *Let $n = \ell k$. Let $F : \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ be a generalized cyclotomic mapping with bijective exponents with respect to $\mathbb{F}_{2^k}$. Let $\beta \in \mathbb{F}_{2^n}^*$ and $\mathcal{K}(\beta) = \left|\{\gamma \in \Gamma : \mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\beta F(\gamma)) = 0\}\right|$. Then:*

$$W_F(0,\beta) = 2^k \left(\mathcal{K}(\beta) - \sum_{i=0}^{\ell-2} 2^{ik}\right).$$

Proof. By Proposition 6.77, $W_F(0,\beta)$ can be expressed as:

$$W_F(0,\beta) = -\sum_{i=1}^{\ell-1} 2^{ik} + \sum_{\gamma\in\Gamma} W_{\mathbb{F}_{2^k},x^{d_\gamma}}\left(0, \mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}\left(\beta F(\gamma)\right)\right).$$

Moreover, $x \mapsto x^{d_\gamma}$ is a bijection over $\mathbb{F}_{2^k}$ and thus, $W_{\mathbb{F}_{2^k},x^{d_\gamma}}(0,\lambda) = 2^k \cdot \mathbf{1}_0(\lambda)$. Then:

$$W_F(0,\beta) = -\sum_{i=1}^{\ell-1} 2^{ik} + 2^k \left|\left\{\gamma \in \Gamma \text{ s.t. } \mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\beta F(\gamma)) = 0\right\}\right|.$$

□

When n is even and $k = n/2$, the Walsh coefficients in zero are directly derived from the number of preimages by F of the multiplicative cosets of $\mathbb{F}_{2^k}$.

Corollary 6.79 (Walsh coefficients in zero when $k = n/2$)**.** *Let $n = 2k$ be an even integer and $F : \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ be a generalized cyclotomic mapping with bijective exponents with respect to $\mathbb{F}_{2^k}$. Then, for any $\beta \in \mathbb{F}_{2^n}^*$, we have:*

$$W_F(0,\beta) = 2^k(\mathcal{K}(\beta) - 1), \quad \text{with} \quad \mathcal{K}(\beta) = \left|\Gamma \cap F^{-1}(\beta^{-1}\mathbb{F}_{2^k})\right|.$$

Most notably, if F is a plateaued APN function, there are at least $\frac{2(2^k+1)}{3}$ cosets of $\mathbb{F}_{2^k}^*$ with 0 or 2 preimages by F.

Proof. We know from Proposition 6.78 that $W_F(0,\beta) = 2^k(\mathcal{K}(\beta) - 1)$. Since $k = n/2$, $\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\beta F(\gamma)) = \beta F(\gamma) + (\beta F(\gamma))^{2^k} = 0$ if and only if $\beta F(\gamma) \in \mathbb{F}_{2^k}$, i.e., $F(\gamma) \in \beta^{-1}\mathbb{F}_{2^k}$. We deduce that: $\mathcal{K}(\beta) = |\Gamma \cap F^{-1}(\beta^{-1}\mathbb{F}_{2^k})|$. If F is APN then by Proposition 6.74, $F^{-1}(\{0\}) = \{0\}$. In that case this implies that: $\mathcal{K}(\beta) = \left|\Gamma \cap F^{-1}(\beta^{-1}\mathbb{F}_{2^k}^*)\right|$, which is the number of cosets of $\mathbb{F}_{2^k}^*$ mapped onto $\beta^{-1}\mathbb{F}_{2^k}^*$. □

Corollary 6.80. *Let $n = 2k$ be an even integer and $F : \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ be a generalized cyclotomic mapping with bijective exponents with respect to $\mathbb{F}_{2^k}$. Then its linearity satisfies:*

$$\mathcal{L}(F) \geq 2^k \left(\max_{\beta\in\Gamma}(\left|\Gamma \cap F^{-1}(\beta^{-1}\mathbb{F}_{2^k})\right| - 1\right).$$

Example 6.81. We exhaustively looked at the 63 mappings over $\mathbb{F}_{64}$ of the form $x \mapsto x^3 + x^{10} + ux^{24}$, where $u \neq 0$. Eight of them can be proven non-APN thanks to Proposition 6.74, because a coset is set onto $\{0\}$. For the remaining ones, we computed the multiset $\{\!\{\mathcal{K}(\beta),\ \beta \in \Gamma\}\!\}$ where Γ is a system of representatives of the multiplicative cosets of $\mathbb{F}_{2^k}^*$. The possible values for this multiset are:

$$\begin{aligned}
M_1 &= \{\!\{0,\ 0,\ 1,\ 1,\ 1,\ 1,\ 1,\ 2,\ 2\}\!\},\\
M_2 &= \{\!\{0,\ 0,\ 0,\ 0,\ 1,\ 1,\ 1,\ 3,\ 3\}\!\},\\
M_3 &= \{\!\{0,\ 0,\ 0,\ 1,\ 1,\ 1,\ 1,\ 2,\ 3\}\!\},\\
M_4 &= \{\!\{0,\ 0,\ 0,\ 0,\ 1,\ 1,\ 2,\ 2,\ 3\}\!\},\\
M_5 &= \{\!\{0,\ 0,\ 1,\ 1,\ 1,\ 1,\ 1,\ 1,\ 3\}\!\},\\
M_6 &= \{\!\{1,\ 1,\ 1,\ 1,\ 1,\ 1,\ 1,\ 1,\ 1\}\!\},
\end{aligned}$$

In this specific case, it holds that $\frac{2(2^k+1)}{3} = \frac{2\times 9}{3} = 6$. This implies that if a function has as spectrum M_1, M_2, M_3, M_5 or M_6, then it cannot be APN. The spectrum M_4 cannot be rule out. In practice, it is obtained for the six roots u of $X^6+X^4+X^3+X+1$ and the associated functions are in that case (CCZ-equivalent to) the Kim mapping, and therefore APN. ▷

The functions considered in this example are quadratic cyclotomic mappings with respect to the subfield $\mathbb{F}_{2^{\frac{n}{2}}} \subset \mathbb{F}_{2^n}$, with n even. As detailed below and due to a recent result of Göloğlu [Göl23], no new APN functions can be found in this family. However, the previous results are more general and could hopefully lead to the finding of new APN functions outside of this specific family.

6.5.3 APN cyclotomic mappings of degree 2

As already mentioned, quadratic APN functions are relatively better understood than the general case. It is therefore interesting to look at this subcase in the context of cyclotomic mappings.

First of all, as already mentioned by Göloğlu [Göl15, p.264] in a less general case, quadratic cyclotomic mappings with respect to subfields can easily be characterized by refining Lemma 6.14.

Proposition 6.82 (Quadratic cyclotomic mappings w.r.t subfields)**.** *Let $\mathbb{F}_{2^k} \subset \mathbb{F}_{2^n}$. Let $d < 2^k - 1$. Let $F\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ be a quadratic cyclotomic mapping of exponent d with respect to $\mathbb{F}_{2^k}$. Then, $\mathrm{wt}(d) \leq 2$. Furthermore if $\mathrm{wt}(d) = 2$ with $d = 2^{e_1} + 2^{e_2}$, $e_1 \neq e_2$ then F is of the form:*

$$F\colon x \mapsto \sum_{i=0}^{\ell-1}\sum_{j=0}^{\ell-1} \lambda_{i,j} x^{2^{ki+e_1}+2^{kj+e_2}}, \qquad \textit{for some } \lambda_{i,j} \in \mathbb{F}_{2^n}.$$

Proof. Let $(2^k-1)s+d = 2^u+2^v$ be an exponent of weight exactly 2, which appears in the univariate form of F. By Lemma 6.14 we get $d \equiv 2^{u'} + 2^{v'} \bmod 2^k - 1$ where u', v' are the Euclidean remainders of u and v modulo k. If $u' = v' = k-1$ then

$d \equiv 2^k \equiv 1 \bmod 2^k - 1$ and therefore $d = 1$. Otherwise, $2^{u'} + 2^{v'} < 2^k - 1$ and $d = 2^{u'} + 2^{v'}$ which is of weight at most 2.

In the case where $\mathrm{wt}(d) = 2$, F does not contain any linear monomial x^{2^w} as it would imply $d = 2^{w'}$ (where $w' \equiv w \bmod k$), which is a contradiction. Furthermore $F(0) = 0$ so all monomials of F have degree exactly 2. In that case, for any term $x^{2^u+2^v}$ in F, we get $d \equiv 2^{u'} + 2^{v'} \bmod 2^k - 1$ with $u' \neq v'$ (otherwise $\mathrm{wt}(d) = 1$, which is excluded). Since $2^{u'} + 2^{v'} < 2^k - 1$, we deduce $2^{e_1} + 2^{e_2} := d = 2^{u'} + 2^{v'}$. This means that there exist i, j such that $\{u, v\} = \{ki + e_1, kj + e_2\}$. But u, v are such that $2^u + 2^v < 2^{\ell k} - 1$ so necessarily $i, j \in [\![0, \ell - 1]\!]$. □

Lemma 6.75 implies that the case where $\mathrm{wt}(d) = 1$ is not interesting if we are looking for APN cyclotomic mappings, as $x \mapsto x^d$ is linear in that case. When $\mathrm{wt}(d) = 2$, we can always write $d = 2^a(2^s + 1)$ but a can be arbitrarily set to 0. Indeed, $F = x^{2^a(2^s+1)}P(x^{2^k-1})$ is APN if and only if the linearly-equivalent cyclotomic mapping $F(x^{2^{n-a}}) = x^{2^s+1}Q(x^{2^k-1})$ is APN where $Q = P(x^{2^{n-a}})$. We then deduce the following corollary.

Corollary 6.83 (Quadratic APN cyclotomic mappings and Gold exponents). *Studying APN cyclotomic mappings whose exponent is a Gold exponent over $\mathbb{F}_{2^k}$ $d = 2^s + 1$, $\gcd(s, k) = 1$ is sufficient to study quadratic APN cyclotomic mappings with respect to $\mathbb{F}_{2^k}$.*

For even n, the family of quadratic cyclotomic mappings with respect to $\mathbb{F}_{2^{\frac{n}{2}}}$, described in Proposition 6.82, has already received a lot of attention [Göl15, Car15, Bud+17, Li+21, CL21, Göl23]. As shown in Proposition 6.82, they also include the so-called "Kim-type" functions introduced by Carlet in [Car15, Section 3.7] who raised the question of the existence of APN functions in this family. For instance, it contains the APN trinomials for $n \equiv 0 \bmod 4, k = n/2$ that were exhibited by Göloğlu [Göl15], and which have been latter proved affine equivalent to the Gold power mapping $x^{2^{t-i}+1}$ [Bud+17, Section 4].

Most notably, the list of all quadratic APN cyclotomic mappings w.r.t $\mathbb{F}_{2^{\frac{n}{2}}}$, n even, is now known to be complete. Cyclotomic mappings of exponent 3 are all affine-equivalent to either x^3 or $x^{2^{k-1}+1}$ [CL21, Li+21], and thus never CCZ-equivalent to a permutation [GL20]. The general case, with exponent $d = 2^s+1$, has been recently classified by Göloğlu [Göl23], as he classified all APN $(2^s + 1, 2^s + 1)$-projective mappings, which coincide with quadratic cyclotomic mappings with respect to $\mathbb{F}_{2^{\frac{n}{2}}}$ (see Proposition 6.20). His result shows that a quadratic cyclotomic mapping with respect to the subfield $\mathbb{F}_{2^k} \subset \mathbb{F}_{2^n}$ where $n = 2k$ is APN if and only if it is equivalent to some specific Gold mapping (depending on the parities of k and s) *except* when $n = 6$, where it can also be equivalent to the Kim mapping.

This implies that there is no hope to find new APN functions over $\mathbb{F}_{2^n}$, n even, among the quadratic cyclotomic mappings w.r.t $\mathbb{F}_{2^{\frac{n}{2}}}$. However, some families presented in Tables 6.2 and 6.3 contain functions that are equivalent to cyclotomic mappings but not to Gold mappings. This is the case for instance of Families (LK23a) and (LK23b) in Table 6.3, which consist of quadratic cyclotomic mappings with respect to $\mathbb{F}_{2^{\frac{n}{3}}}$ for n divisible by 3. This also applies to more general biprojective mappings. Another interesting idea would be to try to build cyclotomic mappings $\mathbb{F}_{2^k} \subset \mathbb{F}_{2^{2k}}$, with non-quadratic APN exponents, corresponding to other APN monomial functions. The results that we just presented in this section open the door to new research directions as they are more general than the case $n = 2k$, with F quadratic.

Similarly to Proposition 6.82, we provide the generic form of the polynomials corresponding to quadratic ℓ-variate projective mappings, based on the following lemma.

Lemma 6.84 (Quadratic multivariate functions). *Let $F \colon \mathbb{F}_{2^k}^\ell \to \mathbb{F}_{2^k}^\ell$ be quadratic. Then each multivariate monomial in the coordinates of F is of the form $X_i^{u_i} X_j^{u_j}$ with $i \neq j$ and $\mathrm{wt}(u_i) = \mathrm{wt}(u_j) = 1$, or of the form $X_i^{u_i}$ with $\mathrm{wt}(u_i) \leq 2$.*

Proof. Let $(\alpha_1, \ldots, \alpha_\ell)$ be an $\mathbb{F}_{2^k}$-basis of $\mathbb{F}_{2^n}$ with $n = \ell k$. Let us consider the linearly equivalent function $\widetilde{F}$ defined by:

$$\widetilde{F}(x) = \sum_{i=1}^{\ell} \alpha_i F_i(x_1, \ldots, x_\ell),$$

for any $x = \sum_{i=1}^{\ell} \alpha_i x_i$. By construction, $F_i(x_1, \ldots, x_\ell) = \mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\beta_i F(x))$ for some β_i. The function $x \mapsto F(x)^{2^j}$ contains univariate monomials whose exponents are the ones of F multiplied by 2^j. This transformation does not change the Hamming weight of the exponents so $x \mapsto F(x)^{2^j}$ is of algebraic degree at most 2. Moreover, we observe that $(\sum_{i=1}^{\ell} \alpha_i x_i)^{2^a} = \sum_{i=1}^{\ell} \alpha_i^{2^a} x_i^{2^a}$, so a linear univariate monomial X^{2^a} can only produce multivariate monomials with a single variable, and whose exponent is of Hamming weight 1. Similarly, we observe that:

$$\left(\sum_{i=1}^{\ell} \alpha_i x_i\right)^{2^a+2^b} = \sum_{i,j=1}^{\ell} \alpha_i^{2^a} x_i^{2^a} \cdot \alpha_j^{2^b} x_j^{2^b},$$

so a quadratic univariate monomial $X^{2^a+2^b}$ only produces multivariate monomials $X_i^{2^a} X_j^{2^b}$ with $i \neq j$ or monomials $X_i^{2^a+2^b}$. □

As a direct consequence, we obtain the following proposition.

Proposition 6.85 (Quadratic ℓ-variate projective mappings). *Let $F \colon \mathbb{F}_{2^k}^\ell \to \mathbb{F}_{2^k}^\ell$ be a quadratic ℓ-variate projective mappings. Then its exponents $(d_1, \ldots, d_\ell)$ satisfy $\mathrm{wt}(d_i) \leq 2$ for all i. Moreover, in a homogeneous coordinate of exponent $2^s + 1$, only the terms $X_i X_j^{2^s}$ with $i \neq j$ and $X_i^{2^s+1}$ can appear. Most notably, the family of 2-variate projective mappings of exponents $(2^r + 1, 2^s + 1)$ with respect to $\mathbb{F}_{2^k}$ with algebraic degree 2 coincides with the family of $(2^r, 2^s)$-biprojective mappings defined in Definition 6.19.*

Propositions 6.82 and 6.85 then allow the search for new quadratic APN ℓ-variate projective mappings, $\ell > 2$, from their polynomial representations. We believe that this opens a promising direction for finding new APN mappings.

6.6 A deeper analysis of the only known solution to the big APN problem

In this last section, we focus on the study of the Kim mapping. Because it is cyclotomic, symmetries appear in its LAT according to Corollary 6.52, and the number of occurrences of a value in its Walsh spectrum is a multiple of $|\mathbb{G}|$, where $\mathbb{G} = \mathbb{F}_{2^3}^* \subset \mathbb{F}_{2^6}^*$ in that case. But it is well-known [Bro+10, CP19] that the *zero* coefficients are heavily related to the CCZ-equivalence, and in particular to CCZ-equivalence with a bijection. After recalling the known results, we give a characterization of the appearance of a $\mathbb{F}_{2^{\frac{n}{2}}}$-space of zeroes in the LAT of a cyclotomic mapping over $\mathbb{F}_{2^n}$ with respect to $\mathbb{F}_{2^{\frac{n}{2}}}$, just as in the LAT of the Kim mapping. Hopefully, this result could lead to a better understanding of this noteworthy phenomenon, and of the closely-related big APN problem.

6.6.1 CCZ equivalence and Walsh zeroes

Recall that the *Kim mapping* [Bro+10] is the quadratic APN function of 6 variables defined by:

$$\kappa \colon \mathbb{F}_{64} \to \mathbb{F}_{64} \quad x \mapsto x^3 + x^{10} + ux^{24},$$

where u is a root of the primitive polynomial $X^6 + X^4 + X^3 + X + 1$. The most remarkable property of κ is that it is CCZ-equivalent to a permutation. In particular, it led to the only known solution (up to equivalence) to the big APN problem, which is known as the *Dillon permutation*.

In order to present our characterization of this phenomenon, we recall the definition of the *Walsh zeroes*.

Definition 6.86 (Walsh zeroes)**.** Let $F \colon \mathbb{F}_2^n \to \mathbb{F}_2^m$. The Walsh zeroes of F are the element of the set:

$$\mathcal{Z}_F := \{(\alpha, \beta) \in \mathbb{F}_2^n \times \mathbb{F}_2^m,\ W_F(\alpha, \beta) = 0\} \cup \{(0,0)\}\,.$$

In other words, the Walsh zeroes are the preimages of 0 by the Walsh transform, to which we add $(0, 0)$. ▷

The Walsh zeroes of a function enable us to obtain another characterization of admissible mappings, which, as defined in Definition 2.61, are the mappings that lead to a CCZ-equivalence relation between two functions.

Proposition 6.87 (Second characterization of admissibility [CP19, Theorem 1])**.** *Let $F \colon \mathbb{F}_2^n \to \mathbb{F}_2^m$. Let $\mathcal{A}$ be an affine mapping over $\mathbb{F}_2^{n+m}$ with linear part $\mathcal{L}$. Then $\mathcal{A}$ is admissible for F if and only if $\mathcal{L}(\mathbb{F}_2^n \times \{0\}) \subset \mathcal{Z}_F$.*

Proof. According to Lemma 2.62, $\mathcal{A}$ is admissible if and only if the function $x \mapsto \mathcal{A}_1(x, F(x))$ is bijective. However a function is bijective if and only if all its non-zero components are balanced. In other words, $\mathcal{A}$ is admissible if and only if, for any $a \in \mathbb{F}_2^n \setminus \{0\}$, it satisfies:

$$\begin{aligned} 0 = \sum_{x \in \mathbb{F}_2^n} (-1)^{a \cdot \mathcal{A}_1(x,F(x))} &\iff 0 = \sum_{x \in \mathbb{F}_2^n} (-1)^{(a,0) \cdot \mathcal{A}(x,F(x))} \\ &\iff 0 = (-1)^{(a,0)\cdot\mathcal{A}(0,0)} \sum_{x \in \mathbb{F}_2^n} (-1)^{(a,0)\cdot\mathcal{L}(x,F(x))} \\ &\iff 0 = (-1)^{(a,0)\cdot\mathcal{A}(0,0)} \sum_{x \in \mathbb{F}_2^n} (-1)^{\mathcal{L}^\top(a,0)\cdot(x,F(x))} \\ &\iff \mathcal{L}^\top(a,0) \in \mathcal{Z}_F. \end{aligned}$$

Because $(0,0)$ is by definition an element of $\mathcal{Z}_F$, this is equivalent to state that $\mathcal{L}(\mathbb{F}_2^n \times \{0\}) \subset \mathcal{Z}_F$. □

Corollary 6.88. *A mapping $\mathcal{A}$ is admissible for a function F if and only if its linear part $\mathcal{L}$ is admissible for F.*

From this characterization, we deduce the following important characterization of functions that are CCZ-equivalent to a bijection.

Corollary 6.89 (CCZ equivalence and Walsh zeroes [Bro+10]). *Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$. Then F is CCZ-equivalent to a bijection if and only if there exist two $\mathbb{F}_2$-spaces of dimension n, $V, W \in \mathcal{Z}_F$ which satisfy $V \cap W = \{0\}$.*

Proof. The necessary condition is a consequence of Proposition 2.64: indeed up to CCZ equivalence, we have $|W_F(\alpha,\beta)| = \left|W_G(\mathcal{L}^\top(\alpha,\beta))\right|$, where $\mathcal{L}$ is the linear part of the affine function mapping one graph to the other. But as $\mathcal{L}^\top$ is linear and bijective, it maps vector spaces of zeroes onto vector spaces of zeroes of the same dimension and it also preserves the dimension of the intersection. In particular the space $\mathbb{F}_2^n \times \{0\}$ is always a subspace of $\mathcal{Z}_F$ (because the linear functions are balanced) and $\{0\} \times \mathbb{F}_2^n$ belongs to $\mathcal{Z}_F$ if and only F is bijective, which is the case here by hypothesis.

Conversely, let V, W be two spaces of dimension n, which are in direct sum. By choosing a basis (v_i), (w_i) of both spaces, we obtain a basis of the full space by concatenating them. In particular, there exists (a unique) bijective linear function $L\colon \mathbb{F}_2^{2n} \to \mathbb{F}_2^{2n}$ which maps for any $i \in [\![1,n]\!]$, $\xi^{(i)}$ to v_i and $\xi^{(n+i)}$ to w_i. The function L satisfies: $L(\mathbb{F}_2^n \times \{0\}) = V \subset \mathcal{Z}_F$, so $L^\top$ defines an admissible mapping for F. Furthermore, the associated function G satisfies $\{0\} \times \mathbb{F}_2^n = L^{-1}(W) \subset \mathcal{Z}_G$. This implies that G is a bijection. □

Because the Kim mapping is CCZ equivalent to a bijection, Corollary 6.89 implies that the set of the Walsh zeroes of the Kim mapping contains two such $\mathbb{F}_2$-subspaces.[4] But the function actually satisfies a strictly stronger condition.

[4] This is precisely this argument that is used in [Bro+10] to find the APN bijection in dimension 6.

Indeed, because the Kim mapping is defined over $\mathbb{F}_{64}$, it is easier (but equivalent) to deal with $\mathcal{Z}_F \subset \mathbb{F}_{64} \times \mathbb{F}_{64}$ (rather than $\mathcal{Z}_F \subset \mathbb{F}_2^6 \times \mathbb{F}_2^6$). In that case, it is possible to find two spaces V, W of the form $\alpha\mathbb{F}_8 \times \beta\mathbb{F}_8$ which are in direct sum. Such spaces are $\mathbb{F}_2$-spaces of dimension $n = 6$ but they are even $\mathbb{F}_8$-spaces of dimension 2. With the point of view taken since the beginning of this chapter, the structure of theses spaces is not *that* surprising (while their sole existence is !). Indeed, it is reminiscent of the symmetries of the Walsh transform of cyclotomic mappings mentioned in Lemma 6.50 and Corollary 6.52.

Stated otherwise, in the case of a cyclotomic mapping F with bijective exponent, such a space $\alpha\mathbb{F}_{2^{\frac{n}{2}}} \times \beta\mathbb{F}_{2^{\frac{n}{2}}}$ is a subset of the Walsh zeroes if and only if $W_F(\alpha\varphi, \beta) = 0$ for all $\varphi \in \mathbb{F}_{2^{\frac{n}{2}}}$, because the other 0 are spread out by the symmetries. This is also equivalent to $W_F(\alpha, \beta\varphi) = 0$ for all $\varphi \in \mathbb{F}_{2^{\frac{n}{2}}}$.

With the benefit of hindsight, these $\mathbb{F}_8$-spaces of zeroes clearly appear in Figure 6.2 as 7 × 7 grey squares, which must be completed using the very first column and row which correspond to $\alpha = 0$, $\beta = 0$. Two of them can be chosen to have a trivial intersection: to do so they must lie neither in the same “row” nor “column”, where row/column refer here to a set of 7 consecutive rows or columns.

6.6.2 $\mathbb{F}_{2^{\frac{n}{2}}}$-subspaces in the Walsh zeroes of cyclotomic mappings

The following proposition and theorem provide a necessary and sufficient condition for the existence of such (α, β) when F is a cyclotomic mapping with respect to $\mathbb{F}_{2^{\frac{n}{2}}}$.

Proposition 6.90 (Trivial square in the Walsh zeroes). *Let $n = 2k$. Let $F : \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ be a cyclotomic mapping of bijective exponent d with respect to $\mathbb{F}_{2^k}$. Let us suppose that $F(\mathbb{F}_{2^n}) = \beta^{-1}\mathbb{F}_{2^k}$, for some β. Then $W(\alpha, \beta) = 0$ for any $\alpha \in \mathbb{F}_{2^n}^*$.*

Proof. By hypothesis for any $\lambda \in \mathbb{F}_{2^n}$, $F(\lambda) \in \beta^{-1}\mathbb{F}_{2^k}$, *i.e.* $\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\beta F(\lambda)) = 0$. In that case, by Proposition 6.77, we get:

$$\begin{aligned} W_{\mathbb{F}_{2^n},F}(\alpha, \beta) &= -2^k + \sum_{\gamma \in \Gamma} W_{\mathbb{F},x^d}(\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\alpha\gamma), 0) \\ &= -2^k + \sum_{\gamma \in \Gamma} 2^k \cdot \mathbf{1}_0(\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\alpha\gamma)), \end{aligned}$$

where we use the bijectivity of $x \mapsto x^d$ for the second equality. But Γ is a system of representatives of representatives of the multiplicative cosets of $\mathbb{F}_{2^k}$., so $\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\alpha\gamma) = 0$ has a unique solution γ in Γ, because $\alpha \neq 0$. Thus $W_{\mathbb{F}_{2^n},F}(\alpha, \beta) = -2^k + 2^k = 0$. □

The previous proposition trivially implies that, for the appropriate β and all $\alpha \in \mathbb{F}_{2^n}^*$, the Walsh zeroes contain the spaces $\alpha\mathbb{F}_{2^k} \times \beta\mathbb{F}_{2^k}$. Unfortunately, the following theorem, which characterizes the non-trivial Walsh zeroes subspaces of this form, implies that no other space *of the specific form* $\alpha\mathbb{F}_{2^k} \times \beta\mathbb{F}_{2^k}$ can be found for such F. Note also that this is a very degenerate case where $|Im(F)| =$

$2^k < 2^k + 1 = \frac{2^n-1}{2^k-1} < \frac{2^n-1}{3}$ as soon as $k > 2$, so such a function cannot be APN, due to Corollary 6.70.

Theorem 6.91. *Let $n = 2k$. Let $F : \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ be a cyclotomic mapping of bijective exponent d with respect to $\mathbb{F}_{2^k}$, and Γ be a system of representatives of the multiplicative cosets of $\mathbb{F}_{2^k}$. Let $(\alpha, \beta) \in \mathbb{F}_{2^n}^* \times \mathbb{F}_{2^n}^*$ such that $F(\mathbb{F}_{2^n}) \neq \beta^{-1}\mathbb{F}_{2^k}$ and let G be defined by:*

$$G \colon \Gamma \setminus \Gamma_{\beta^{-1}} \to \mathbb{F}_{2^k} \qquad \gamma \mapsto \frac{\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\alpha\gamma)}{\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\beta F(\gamma))^e};$$

where $\Gamma_{\beta^{-1}} =: \{\gamma \in \Gamma : \mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\beta F(\gamma)) = 0\}$ and $x \mapsto x^e$ is the inverse of $x \mapsto x^d$ over $\mathbb{F}_{2^k}$. Then,

$$W_{\mathbb{F}_{2^n},F}(\alpha\varphi_1, \beta\varphi_2) = 0, \ \forall \varphi_1, \varphi_2 \in \mathbb{F}_{2^k}^* \tag{6.10}$$

if and only if G is bijective.

Remark 6.92. Note that $G(\gamma) = 0 \iff \mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\alpha\gamma) = 0$, which means that $G(\gamma) = 0$ has at most 1 solution: the unique $\gamma_{\alpha^{-1}} \in \Gamma \cap \alpha^{-1}\mathbb{F}_{2^k}$, (if $\gamma_{\alpha^{-1}} \notin \Gamma_{\beta^{-1}}$). ▷

Proof. First of all, $F(\mathbb{F}_{2^n}) \neq \beta^{-1}\mathbb{F}_{2^k}$ which implies that $\beta F(\mathbb{F}_{2^n}) \neq \mathbb{F}_{2^k}$. Therefore the domain of G is not empty, and G is well defined. We decompose the following proof into several steps.

(i) Notation. For any $u \in \mathbb{F}_{2^k}$, we denote the sizes of preimages by:

$$C_u := |\{\gamma \in \Gamma \setminus \Gamma_{\beta^{-1}} : G(\gamma) = u\}|.$$

We also shorten $W_u := W_{\mathbb{F}_{2^n},F}(\alpha u, \beta)$, and denote the sequence of W_u by: $W := (W_v)_{v \in \mathbb{F}_{2^k}}$. We denote the Fourier transform of W by:

$$\widehat{W} := (\widehat{W}_u)_{u \in \mathbb{F}_{2^k}}, \quad \widehat{W}_u := \sum_{v \in \mathbb{F}_{2^k}} (-1)^{uv} W_v.$$

(ii) Rewriting W_u. Thanks to Proposition 6.77 and to the symmetries of the Walsh coefficients of $x \mapsto x^d$, we notice that, for all $u \in \mathbb{F}_{2^k}^*$,

$$W_u = -2^k + S + \sum_{\gamma \in \Gamma \setminus \Gamma_{\beta^{-1}}} W_{\mathbb{F},x^d}(uG(\gamma), 1); \tag{6.11}$$

where $S := \sum_{\gamma \in \Gamma_{\beta^{-1}}} W_{\mathbb{F}_{2^k},x^d}\left(u\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\alpha\gamma), 0\right)$. But as $u \neq 0$, the equation $u\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\alpha\gamma) = 0$ has a single solution $\gamma_{\alpha^{-1}}$ in Γ. If $\gamma_{\alpha^{-1}} \in \Gamma \setminus \Gamma_{\beta^{-1}}$, then $G(\gamma_{\alpha^{-1}}) = 0$, $C_0 = 1$, but also $S = 0$. Otherwise, $C_0 = 0$ and $S = 2^k$. Thus, $S = 2^k(1 - C_0)$ and Eq. (6.11) becomes:

$$\begin{aligned} W_u &= -2^k C_0 + \sum_{\gamma \in \Gamma \setminus \Gamma_{\beta^{-1}}} W_{\mathbb{F}_{2^k},x^d}(uG(\gamma), 1) \\ &= -2^k C_0 + \sum_{y \in \mathbb{F}_{2^k}} C_y W_{\mathbb{F}_{2^k},x^d}(uy, 1). \end{aligned} \tag{6.12}$$

The sum in Eq. (6.12) is actually a sum over $\mathbb{F}_{2^k}^*$ as $W_{\mathbb{F}_{2^k},x^d}(0,1) = 0$.

(iii) Rewriting $\widehat{W}_v$. Let $v \in \mathbb{F}_{2^k}$. Using Eq. (6.12) in the definition of $\widehat{W}_v$, we get

$$\begin{aligned}\widehat{W}_v &= W_0 - 2^k C_0 \sum_{u\in\mathbb{F}^*_{2^k}} (-1)^{uv} + \sum_{u\in\mathbb{F}^*_{2^k}} (-1)^{uv} \sum_{y\in\mathbb{F}^*_{2^k}} C_y W_{\mathbb{F}_{2^k},x^d}(uy,1)\\ &= W_0 - 2^k C_0(2^k \cdot \mathbf{1}_0(v) - 1) + \sum_{y\in\mathbb{F}^*_{2^k}} C_y \sum_{z\in\mathbb{F}_{2^k}} (-1)^{z^d} \sum_{u\in\mathbb{F}^*} (-1)^{u(yz+v)}\\ &= W_0 - 2^k C_0(2^k \cdot \mathbf{1}_0(v) - 1) + \sum_{y\in\mathbb{F}^*_{2^k}} C_y \sum_{z\in\mathbb{F}_{2^k}} (-1)^{z^d} \Big(2^k \cdot \mathbf{1}_0(yz+v) - 1\Big);\end{aligned}$$

where we successively developed $W_{\mathbb{F}_{2^k},x^d}(uy,1)$, interchanged the three sums, and used the average of (non)-trivial characters. Because $2^k\mathbf{1}_0(v) - 1 = \sum_{y\in\mathbb{F}^*_{2^k}} (-1)^{v^d y^{-d}}$, but also $\sum_{z\in\mathbb{F}_{2^k}} (-1)^{z^d} = 0$, and $\sum_{z\in\mathbb{F}_{2^k}} (-1)^{z^d}\mathbf{1}_0(yz+v) = 2^k(-1)^{v^d y^{-d}}$, we obtain,

$$\begin{aligned}\widehat{W}_v &= W_0 + \sum_{y\in\mathbb{F}^*_{2^k}} 2^k(C_y - C_0)(-1)^{v^d y^{-d}}\\ &= W_0 + \sum_{y\in\mathbb{F}^*_{2^k}} 2^k(C_{y^{-e}} - C_0)(-1)^{v^d y}. \qquad (6.13)\end{aligned}$$

Defining $A = (A_u)_{u\in\mathbb{F}_{2^k}}$ as: $A_0 = W_0$, and $A_u = 2^k(C_{u^{-e}} - C_0)$ for any $u \in \mathbb{F}^*_{2^k}$, we can restate Eq. (6.13) as:

$$\widehat{W}_v = \widehat{A}_{v^d} \quad \forall v \in \mathbb{F}_{2^k}. \qquad (6.14)$$

(iv) The actual proof. We prove the theorem using the following equivalence given by the symmetries in the Walsh spectrum (see Corollary 6.52):

$$\textit{Eq. } (6.10) \iff \big[W_u = 0, \text{ for all } u \in \mathbb{F}^*_{2^k}\big].$$

Let us suppose that G is bijective. Therefore, we obtain:

$$(2^k + 1) - N_{F,\beta^{-1}} = \Big|\Gamma \setminus \Gamma_{\beta^{-1}}\Big| = |\mathbb{F}_{2^k}| = 2^k,$$

that is, $N_{F,\beta^{-1}} = 1$, which, with Corollary 6.79, gives $W_0 = 0$. The bijectivity of G also means that $C_u = 1$ for all $u \in \mathbb{F}_{2^k}$. Thus A is constant and equal to 0, and by the Fourier inverse so is $\widehat{A} = \widehat{W}$ and thus so is W: Eq. (6.10) is satisfied.

Conversely, let us suppose that $W_u = 0$ for any $u \in \mathbb{F}^*_{2^k}$. In that case, for any $v \in \mathbb{F}_{2^k}$ we have $\widehat{W}_v = \sum_{v\in\mathbb{F}_{2^k}} (-1)^{uv} W_u = W_0(-1)^0 = W_0$ so $\widehat{W}$ is constant and equal to W_0. By Eq. (6.14), since $x \mapsto x^d$ is bijective over $\mathbb{F}_{2^k}$, this also means that $\widehat{A}$ is constant and equal to W_0. So $\widehat{W} = \widehat{A}$, and thus $W = A$. So $0 = A_u = 2^k(C_{u^{-e}} - C_0)$ for any $u \neq 0$. Because $x \mapsto x^{-e}$ is bijective over $\mathbb{F}_{2^k}^*$, we thus get $C_u = C_0$ for any u. But the fact that the preimages partition the domain gives:

$$(2^k+1)C_0 = \sum_{u \in \mathbb{F}_{2^k}} C_u = \left|\Gamma \setminus \Gamma_{\beta^{-1}}\right| = 2^k + 1 - N_{F,\beta^{-1}}.$$

From the preliminary remark, we know that $C_0 \in \{0,1\}$, and $C_0 = 0$ would imply $N_{F,\beta^{-1}} = 2^k + 1$ which is excluded, so $C_0 = 1$. Therefore $C_u = 1$ for all u and G is bijective.

□

Corollary 6.93 (Necessary condition)**.** *Let $n = 2k$ and let $F : \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ be a cyclotomic mapping of bijective exponent d with respect to $\mathbb{F}_{2^k}$, and Γ be a system of representatives. Let $(\alpha, \beta) \in \mathbb{F}_{2^n}^* \times \mathbb{F}_{2^n}^*$ such that $F(\mathbb{F}_{2^n}) \neq \beta^{-1}\mathbb{F}_{2^k}$ and $\alpha\mathbb{F} \times \beta\mathbb{F}$ is a Walsh zeroes subspace. Then $N_{F,\beta^{-1}} = 1$. Moreover, denoting $\{\mu\} := \Gamma_{\beta^{-1}}$, we have $\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\alpha\mu) \neq 0$.*

This theorem provides a characterization which is based on the values of F rather than its Walsh spectrum. It should be further investigated. In particular, the component functions $x \mapsto \mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(\beta F(x))$ are well understood as they are necessarily homogeneous functions of exponent d because of the multivariate characterization of cyclotomy given in Theorem 6.18.

While this is left as future work, it should be noted that composing a homogeneous function of exponent d with the function $x \mapsto x^{-e}$ where $ed \equiv 1 \bmod 2^k - 1$ is very intriguing. It is likely that this expression could be simplified and could lead to a more precise understanding of this property.

Furthermore, in the very specific case of the Kim mapping $-e = 2$. This implies that $x \mapsto x^{-e}$ is in that case the Frobenius automorphism. Even the less precise fact that it is linear is *a priori* very different from the general case. Could this explain the very peculiar behavior of the Kim mapping ?

Finally, the problem of determining whether a quadratic APN cyclotomic mapping with respect to $\mathbb{F}_{2^{\frac{n}{2}}}$ is CCZ-equivalent to a bijection is closed by Göloğlu [Göl23] as a corollary of his classification: such a mapping is CCZ-equivalent to a bijection if and only if it is equivalent to the Kim mapping. However, our result does not (explicitly at least) take into account the degree of the function and therefore opens the question of whether non-quadratic cyclotomic mappings could be CCZ-equivalent to a bijection.

6.7 Concluding remarks

As shown in this chapter, the study of APN functions is very rich, and it is clear that an even richer theory still remains to be developed. While this work does not come with new instances of APN function yet, we expect that the theoretical arguments that we developed will benefit future lines of research that are more focused on exhibiting such new examples. To be honest, finding new APN functions was actually our primary goal, but we had to take some side roads.

Necessary tools. A first clear obstacle that we encountered was the lack of publicly-available tools to properly study APN functions. While computational algebra system such as SageMath [Sage24] or Magma are now widespread in the community, a substantial effort is still needed to provide efficient and convenient libraries to tackle some very specific problems. The initiative taken by Léo Perrin with SboxU [Bau+24b] is a first step to fulfill this need, but it still needs some functionalities. Moreover, while some hard-coded lists of APN functions appears in SboxU, it is not sufficient to guarantee that a newly found APN function is indeed new. In fact, the number of known functions is continually growing, and no collaborative database is maintained. As a first step toward such a goal, the author of this thesis took a substantial time to implement all the infinite APN functions listed in Tables 6.2 to 6.4. These implementations are currently based on either the initial univariate or multivariate representations given by their designers. However, at the end of this chapter, it is clear that such a dichotomy has no reason to be and that a more convenient interface has to be designed in order to handle different representations. Hopefully, this will be soon made publicly available. A second goal in this direction would be to build a database of the known APN functions in which, for instance, the numerous invariants of each non-equivalent function could be stored. This work was carried out by many researchers to provide arguments of newness, but a public database would be a more convenient way of storing and accessing, for instance, the Walsh and differential spectra of an ortho-derivative.

Necessary clean up. Another related point is the necessary clean up of the infinite APN classes. From purely computational evidence, it is clear that given an infinite family, many of the parameter sets may lead to equivalent functions. Sometimes, these equivalences are known. For instance, we already mentioned the case of the families (BCL09a/b/c) where the parameter a appears as an element of $\mathbb{F}_{2^n}^*$ in most tables of recent papers, while it is known since the original work [BCL09b, Section II.B] that it can be reduced to one or two values depending on the parity of n. This again is an unnecessary obstacle to the finding of new APN functions, which is already sufficiently hard as it is. This is also the case for the parameter a of the (BHK20) family which can be reduced to a sole value. Furthermore, Kaspers & Zhou [KZ21, KZ22] proved that this is also the case of the parameter a for the family (ZP13) in [KZ21, Theorem 1.1], and that, other than that, any other choices of parameters lead to CCZ-inequivalent functions. They also provide the same kind of results for the family (T19) in [KZ22]. With the results that we present in this chapter, their methodology seem *very promising* to try to reduce the number of representatives in each family. Furthermore, they are able to explicitly determine the cardinality of the automorphism groups for the previous families. This invariant is another useful tool to compare functions from distinct families, and again, avoid to study twice functions that may belong to two distinct families.

Cryptanalysis of an Sbox. Along our path, the point of view that we developed in this chapter and the computational tools that we programed also enabled us to give new insights on a different, yet related, problem. Indeed, we took a look back at previous cryptanalyses [Per19, PU16, BPU16] of the Sbox of two recent Russian standards, namely Streebog and Kuznyechik [Fed12, Fed15]. This Sbox is known to have a surprising structure with respect to both additive and multiplicative cosets of the subfield $\mathbb{F}_{2^{\frac{n}{2}}}$. It of course resonates with this chapter. With the same kind of methodology, we are able to clarify a bit more the choices of the subcomponents of this function. We also generalize its construction, and showcase the interest of such a family by presenting specific instances with surprising properties such as a very-low linearity. This is presented in Chapter B.

Back to the main road. Finally, because of these difficulties, we focused on a theoretical study. Our work enables us to get a better picture of the diversity among the infinite APN families. In particular, while tremendous efforts were made to diversify the known APN functions, it is astonishing to observe that the vast majority shares some very peculiar properties. The fact that most of them are linearly self-equivalent is already intriguing, but not as much as the fact that all these self-equivalences are almost of the same nature. As suggested all along the chapter, this reaffirms some open questions and leads to many others that we recall below.

1. Is an APN function always CCZ self-equivalent ? linearly self-equivalent ?
2. Is a quadratic function always linearly self-equivalent ?
3. Can we find stronger necessary conditions for cyclotomic mappings with respect to subfields/projective mappings to be APN ?
4. Can we find conditions for cyclotomic mappings with respect to subgroups ?
5. Can we find more APN cyclotomic/projective mappings by using smaller subfield than $\mathbb{F}_{2^{\frac{n}{2}}} \subset \mathbb{F}_{2^n}$?
6. Can we use the cyclotomic or projective constructions with APN monomials of higher degree to build new functions inequivalent to quadratic functions ?

7. Can we prove or disprove the existence of linearly self-equivalent representatives among the sporadic examples, and in particular for the Brinckmann-Leander-Edel-Pott cubic and for the functions for $n = 8$ found by QAM methods ?

Chapter 7

Design of symmetric primitives for emerging use cases

Contrary to the four previous chapters, this one is dedicated to the design of symmetric primitives. Of course, designing and cryptanalysing cannot be considered separately, however the latter one comes with many new challenges. In particular, a new primitive should meet a need, and create its own place among all previous ones. The first phase of a design process is wide open, and therefore particularly challenging. But again, while novelty is obviously seeked, it should not be traded for security nor efficiency. This is the reason why the initial cryptanalysis and benchmarks are essential. Furthermore, these first metrics give a starting point to figure out what can and/or must be improved.

Not only is this chapter rather different from the previous ones, it also tackles two use cases and two kinds of primitives that are radically different from the cases of lightweight permutation-based AEAD encryption schemes such as Ascon, or lightweight block ciphers such as Midori that were presented to a great extent in Chapters 3 to 5. However, and as we will see, both designs are still heavily influenced by the design and analysis of block ciphers.

First, we consider the case of message authentication codes (MAC, see Definition 1.4) based on the AES. As already mentioned, the AES is today the most widespread, and arguably the most secure block cipher. It has resisted at least for more than 25 years to intensive cryptanalysis. This is the reason why optimizing its implementation became a very important challenge. As astonishing as it can be, an AES round can today achieves a latency and and a throughput which is comparable to the ones of a few logical XORs on a modern CPU with AES instructions. Though it can be counter intuitive thinking about the involved hardware circuit, this is the case for modern processors supporting the AES-NI [Gue08] set of hardware-accelerated instructions designed by Intel for their modern processors. With such performances, the full AES remains competitive and a line of research studies mode of operations [MV04, RBB03, KR21] to take advantage of this performance boost. But the AES round *alone* became also a possible component for new designs thanks to this instruction set. However, while at least a dozen new primitives tackles different use cases, it is surprising that *dedicated* message authentication codes were not presented before. To take up this challenge, we first focus on the design of *universal hash functions* which is a kind of primitives that comes with its own precise security notions. Then only are our MACs derived by using the EWCDM construction [CS16]. In order to present a competitive family of UHFs, we rely on automated tools. First, we present an automated tool that, given a candidate, builds an associated *Mixed Integer Linear Programming* (MILP) problem which encodes the counting of the number of actives Sboxes. The solving of this problem (thanks to a dedicated solver) then provides an upper bound on the probability of the differential trails. Secondly, we automated the compilation and benchmarking of the studied candidates to automatically measure their performances. In the end, we obtain a UHF that can reach better performances than state-of-the-art ones. We also derive from it a MAC which is, as of today, the fastest one on modern desktop/server processors. This first section is based on a joint work with Augustin Bariant, Gaëtan Leurent, Clara Pernot, Léo Perrin & Thomas Peyrin that is published in the IACR Transactions on Symmetric Cryptology, 2024(2) [Bar+24]. The presentation made in this thesis is a strict subset of the subjects that are detailed in the published paper.

The second part of this chapter is focused on a problem coming from asymmetric cryptography. A so-called *fully homomorphic encryption* (FHE) is an algorithm which enables to execute any kind of computations on both plain and encrypted data *without decrypting the former ones*, and return an encrypted solution. While the effective solutions are quite recent in the history of cryptography, the interest for such technologies is exploding. Indeed, with the growth of cloud providers that enables the user to outsource its computations, such techniques seem necessary to ensure security, especially when sensitive data have to be analyzed. However, the methods that exist today remain very costly in terms of computations, but also in terms of bandwidth. Indeed, and contrary to the case of symmetric ciphertexts, the current FHE ciphertexts are much larger than the size of the original plaintexts. This is the reason why the so-called *transciphering* [NLV11] technique suggests a trade-off which enables the user to exchange less data with the server, at the cost of an increased number of computations on the server side. Instead of directly encrypting its data using the FHE scheme, the user can encrypt them using a symmetric cipher and then send it under this form. On top of that, the user provides the server with an FHE encryption of the secret key that was used to encrypt the data. Because any computation can be homomorphically executed, the homomorphic decryption of the symmetric ciphertexts is possible, and leads to FHE ciphertexts containing the result, *i.e.* FHE ciphertexts associated to the plaintexts. From there, the true outsourced computations can start.

However, the symmetric cipher must be homomorphically evaluated on the server side. Therefore, because of the native constraints of the FHE schemes, the symmetric encryption algorithms that we usually use lead to very costly FHE implementations. This is the reason why dedicated symmetric algorithms are needed for this specific use case. Previous works [Can+16, Méa+16] have highlighted that a *stream cipher* (see Definition 1.8) is far more suitable for this application than a block cipher in CBC mode. In this chapter we then focus on the design of a stream cipher over the field with 17 elements to be used in a FHE scheme called TFHE [Chi+16, Chi+17, Chi+20]. In particular, our cipher is thought of to take full advantage of the so-called *programmable bootstrapping* feature of TFHE and the low cost of linear operations. Its design is inspired from previous stream ciphers, but also from block ciphers. In particular, by borrowing ideas from the AES, such as the square state and the use of an MDS matrix, we are able to provide strong security arguments that are often lacking when it comes to stream ciphers. Such theoretical arguments are also supported by the solving of associated MILP problems, like in the case of the MACs mentioned above. This section is based on a joint work with Sonia Belaïd, Nicolas Bon, Christina Boura, Anne Canteaut, Gaëtan Leurent, Pascal Paillier, Léo Perrin, Matthieu Rivain, Yann Rotella & Samuel Tap that has been submitted to an international conference. The presentation made in this thesis is again, a selection of the subjects detailed in the published paper.

Contents

7.1 Universal hash functions and MACs based on AES

7.1.1 Context and design goals

7.1.1.a AES-based cryptographic schemes

As already mentioned, for instance in Chapter 4, the AES block cipher [DR02] has deeply influenced the design of symmetric-key cryptographic primitives. This trend even accelerated after the introduction in modern CPUs of the hardware-accelerated set of instructions called AES-NI [Gue08], which implements the AES encryption and decryption. To benefit from that potential performance boost, designers continued studying modes of operations which allow an efficient reuse of the full AES [MV04, RBB03, KR21]. Yet, many new cryptographic designs rely on the AES *round function* as a building block, either for hash functions [Ben+08, Ind+08, BD08, GK08], for authenticated encryption schemes [WP14, Nik14, Jea+21, Sak+21, NFI24], for permutations [Iso+23, GM16, Köl+16, Bos+22], or collision resistant building blocks [JN16, Nik17].

Today, hardware acceleration of the round function of AES is widespread in modern computer CPUs and becomes more and more powerful with a reduced latency and an increased throughput. This allows many symmetric primitives to eventually reach throughput performances under 1 c/B, but advances are still needed, especially to handle the impressive throughput range (100 Gbps to 1 Tbps) of the sixth-generation of mobile communication systems (6G).

This is the direction taken by the Authenticated Encryption (AE) algorithm Rocca [Sak+21, Sak+22] and its updated version Rocca-S [NFI24], which is currently the fastest AE on AES-NI platforms and under submission at IETF. The round function framework of Rocca has even been further analysed and optimal round functions (in terms of speed) have been found within the framework [TSI23]. More generally, there has been significant efforts to design symmetric primitives relying on AES rounds, such as AEGIS [WP14], Tiaoxin [Nik14] or Aerion [Bos+22].

7.1.1.b Universal hash functions and message authentication codes

Security notions of UHFs and MACs. In the following, we study the construction of (almost) universal hash functions (UHF) based on AES rounds. A UHF is a family of functions of the form:

$$H = (h_k \colon \mathbb{F}_2^\star \to \mathbb{F}_2^n)_{k \in \mathbb{F}_2^\kappa},$$

where each function, which is indexed by a key k, takes as input an arbitrary-long (but finite) plaintext and maps it to a fixed-length tag. As for the definitions of block cipher, hash function, or MAC, which are given in Section 1.3.1, the definition of UHF is abstracted from security arguments. The two security notions associated to UHF are given in the following definitions.

Definition 7.1 (ε-AU)**.** Let $\varepsilon \in [0,1]$. A family of functions $H = (h_k)_{k \in \mathbb{F}_2^\kappa}$ is *ε-almost-universal* if it satisfies:

$$\forall x, x' \in \mathbb{F}_2^\star, \text{ s.t. } x \neq x', \quad \frac{|\{k \in \mathbb{F}_2^\kappa : h_k(x) = h_k(x')\}|}{2^\kappa} \leq \varepsilon.$$

▷

Definition 7.2 (ε-AXU)**.** Let $\varepsilon \in [0,1]$. A family of functions $H = (h_k \colon \mathbb{F}_2^\star \to \mathbb{F}_2^n)_{k \in \mathbb{F}_2^\kappa}$ is *ε-almost-XOR-universal* if it satisfies:

$$\forall \Delta \in \mathbb{F}_2^n, \ \forall x, x' \in \mathbb{F}_2^\star, \text{ s.t. } x \neq x', \quad \frac{|\{k \in \mathbb{F}_2^\kappa : h_k(x) + h(x') = \Delta\}|}{2^\kappa} \leq \varepsilon.$$

▷

The ε-AU notion only requires collision resistance on average over a random key. The ε-AXU notion is a stronger variant that covers an arbitrary output difference, rather than just collisions, which corresponds to $\Delta = 0$. In particular, if H is an ε-AXU family, it is also an ε-AU family. These security notions are relatively weak, so that they can be fulfilled by purely combinatorial constructions. However, UHF remains very versatile: as we will see, they can for instance be turned into a MAC with a few extra components.

As presented in Definition 1.4, a MAC also processes a message and a secret key to generate a tag[1]. However, is should ensure authenticity and integrity of the message, and therefore it should be hard for an attacker to forge a tag, *i.e.* to generate a valid combination of message/tag without knowledge of the secret key.

More formally, for a key k, a nonce N and a message x, a nonce-based MAC F consists of a signing algorithm $\mathrm{AUTH}_k(x, N)$ that generates a tag T, and a verification algorithm $\mathrm{VER}_k(x, N, T)$ that returns "valid" if $\mathrm{AUTH}_k(x, N) = T$ and "invalid" otherwise. A (q, v, t)-adversary against the nonce-based MAC F is an adversary $\mathcal{A}$ with access to oracles AUTH_k and VER_k, making at most q MAC queries to the AUTH_k oracle, at most v verification queries to VER_K oracle, and

[1] A nonce-based MAC also takes as input a nonce value that should be used more than once.

running in time at most t. We say that $\mathcal{A}$ *forges* if any of its queries to VER_k returns "valid". The *advantage* of $\mathcal{A}$ against the nonce-based MAC security of F is defined by:

$$\mathbf{Adv}_F^{\mathrm{MAC}}(\mathcal{A}) := \mathbb{P}\left[\mathcal{A}^{\mathrm{AUTH}_k,\mathrm{VER}_k} \text{ forges}\right].$$

where k is picked uniformly at random and where $\mathcal{A}$ is not allowed to ask a verification query (x, N, T) to VER_k if a previous query (x, N) to AUTH_x returned T. Note that $\mathcal{A}$ is also not allowed to repeat nonces for AUTH_k, but can repeat them for VER_k.

MAC based on universal hash functions. MACs are classically built from block ciphers, but also from UHF. Notably, GMAC [Dwo07] and Poly1305 [Ber05] are two popular MACs based on UHF which use polynomial evaluation in a finite field as a UHF. They use the Wegman-Carter-Shoup construction [CW77, Sho96] to construct a nonce-based MAC from UHF. However, it only provides $2^{n/2}$ security for a n-bit tag with unique nonces, and fails completely when nonces are repeated. The EWCDM construction [CS16] guarantees a significantly higher security as it was proven [MN17] to provide essentially 2^n security with unique nonces and even $2^{n/2}$ when nonces are repeated.

Arithmetic UHFs. There has been a significant effort to design fast UHF based on arithmetic operations: polynomial hashing methods such as GHASH used in GCM [MV04] or Poly1305 [Ber05], NH in UMAC [Bla+99], etc. These constructions can be quite fast, and have a proven security level. For instance, GHASH only requires a single multiplication and a single addition in $\mathbb{F}_{2^{128}}$ for every 128-bit block of plaintext. This is particularly interesting in environments where instructions enabling fast arithmetic in the finite field of size 2^{128} are provided, which is the case of most modern processors intended for a usage in servers and desktop computers. On the other hand, Poly1305 and UMAC rely on integer multiplication.

AES-based UHFs. Dedicated design strategies for block ciphers and hash functions are well known, but dedicated Universal Hash Functions (UHFs) have received less attention. We thus focus on designing fast UHFs based on the AES round function. An interesting existing example due to Minematsu & Tsunoo [MT06] is known as PC-MAC. The authors consider four rounds of AES as an ε-AXU family with $\varepsilon \approx 1.18 \cdot 2^{-110}$, under the hypothesis that the round keys are independent, and derive a provably secure MAC with 4 AES rounds per 128-bit block of plaintext. Their analysis is based on the Maximum Expected Differential Probability (MEDP) of 4-round AES that is provided by Keliher and Sui [KS07]. Another interesting work is the EliMAC primitive proposed by Dobraunig *et al.* [DMN23], which uses 11 AES rounds per 128-bit message block (7 rounds can be precomputed in an offline phase, leaving 4 in the online phase). We thus aim for fewer than 4 AES rounds per block of message. However, our security arguments will be heuristic, instead of relying on a formal security proof. Our design goal are detailed in the following section.

7.1.1.c Design goals

While several AES-based constructions exist, we identified substantial room for improvement. We then describe here the objectives our family of UHFs is intended to fulfill. The family in itself is described in Section 7.1.2.

AES-based round functions. As already mentioned, many designs take advantage of the round function of AES, as well as the 128-bit XOR operation, to be both secure and efficient, thanks to the AES-NI instructions set in modern processors. This is for example the case of Tiaoxin [Nik14] and AEGIS [WP14], of the AEAD proposals Rocca [Sak+21, Sak+22] and Rocca-S [NFI24] or of the constructions by Jean & Nikolić [JN16] or by Nikolić [Nik17]. These designs aim at minimizing the so-called *rate* [JN16], that is, the number of AES rounds per 128-bit message block. Rocca (during additional-data processing) and one of the schemes of Jean & Nikolić achieve a rate of 2 for 128-bit security. As presented in the following goal, we adopt a similar strategy.

Goal 7.3. *Our ε-AU families should use AES rounds as internal components for high software performance, and preferably at the lowest possible rate.*

Instruction scheduling. Moreover, modern processors can execute several instructions simultaneously, and schedule instructions as soon as the input operands are ready. The several execution units of a CPU are *pipelined*: an instruction takes several cycles to process, but an execution unit can start processing a new instruction at every clock cycle, even if the previous the previous instruction has not returned its result yet. [Int24]

There are two main metrics to measure the performance of an instruction I. First, the *latency* of I is the number of clock cycles between the beginning of I and the output of its result. We denote it by $L(I)$. The *throughput* of I, which is denoted by $T(I)$, is the number of instructions that can be processed in a given amount of time. We usually consider the reciprocal throughput, which is measured in cycles. In practice, the throughput of an instruction I corresponds to the number of *ports* which can process I. In this work, we focus on the AESENC instruction, which computes one round of AES, and on the 128-bit XOR instruction. This constitutes a fair[2] comparison with previous schemes. The latency and throughput of these instructions for some processors are given in Table 7.1.

In practice, we cannot exploit the full throughput of both types of instructions, because they share some common ports. As shown in Table 7.1 for the more modern processors (at the bottom of the table), AESENC is now available on two ports and the 128-bit XOR on three or four, with a non-trivial intersection between the two sets of ports. This implies that constructions with 2 AES-NI instructions

[2] It could be possible to leverage the more recent 512-bit VAESENC instruction which computes 4 AES rounds in parallel, however it would be hard to compare with previous designs which focus on 128-bit instructions. We can expect that future AES-based designs will greatly benefit from such parallelization.

Architecture	Instr	Latency	Throughput	P_0	P_1	P_2	P_3	P_4	P_5	P_6
Intel Haswell	XOR	1	0.33	×	×				×	
	AESENC	7	1						×	
Intel Skylake	XOR	1	0.33	×	×				×	
	AESENC	4	1	×						
Intel Ice Lake	XOR	1	0.33	×	×				x	
	AESENC	3	0.5	×	×					
Intel Tiger Lake	XOR	1	0.33	×	×				×	
	AESENC	3	0.5	×	×					
AMD Zen 1/2/3/4	XOR	1	0.25	×	×	×	×			
	AESENC	4	0.5	×	×					

Table 7.1: Scheduling of AESENC and XOR instructions on modern processors over the different ports [Fog22].

per 128-bit message block require at least 1 cycle per 128-bit message, *i.e.* at least 0.0625 cycles per byte (c/B). This bound of 0.0625 c/B can only be achieved, if the pipelining is favorable. It should be noted that trying to optimize the scheduling, for instance by minimizing the number of XOR (compared to the number of rounds of AES), is heavily-based on the possibilities provided by modern processors. On the other hand, regardless of implementation tricks, a full throughput on current Intel processors will always remain out of reach for some older algorithms, because their rationale were not thought to take advantage these recent enhancements. More details are given in the published paper [Bar+24].

In addition to the throughput analysis, dependency chains also affect the performance of AES-based constructions [Int24, Section 3.2.2]. There are a lot of other subtleties, which are difficult to exhaustively consider. As a general guideline, we state the following goal.

Goal 7.4. *The instruction scheduling in modern processors should be favorable.*

One way to directly evaluate the performance with state-of-the-art instruction-scheduling algorithms is to compile and benchmark candidates on-the-fly. This strategy exploits advanced techniques from compilers (e.g. modern gcc) or processors, and remains future-proof, since it can easily be adapted to future processors.

Goal 7.5. *Our tool should automatize the benchmarking of candidates. The automatic benchmarking should be adaptable to all processors.*

Security. As we first focus on the design of a UHF family with the ε-AU security, we are only interested in collision resistance. In order to facilitate the security analysis of our candidates, we consider that the output of one of our ε-AU UHFs is not of a single word, but rather an entire state composed of multiple 128-bit words. In addition, we consider that the inner state of the construction is fully unknown, key-dependent, and of full entropy, so that values of the inner states cannot be exploited to build collisions. Thus, in order to ensure collision resistance, it is sufficient in our case to prevent the existence of high-probability differentials of the form $h_k(x) + h_k(x + \Delta) = 0$. We then rely on the following assumption to investigate these.

Assumption 7.6. *The highest probability of a differential trail is a good indication of the highest probability of a differential.*

Thanks to Assumption 7.6, estimating the security level can be done by modeling the differential propagation with a Mixed Integer Linear Programing (MILP) model. This is now a widespread practice [JN16, Sak+21], which is addressed in Section 7.1.4.

Goal 7.7. *A lower bound on the number of active Sboxes in the differential trails of a candidate should be easily computable with computer-aided tools, such as MILP solvers.*

Sum up. Our goals are in line with previous works [JN16, Sak+21]: we want a primitive that favors parallel AES calls to optimize scheduling. This implies that the number of 128-bit XORs should be considered[3] and in fact minimized. As a consequence, we limit ourselves to sparse linear layers.

To compensate the slower diffusion of the sparse linear layer, we consider more sophisticated injection techniques inspired by the design of (tweak-)key schedules. This *a priori* increases the cost of each round (in particular the memory), but it enables the safe use of very simple round functions.

The overall structure is similar to that of PANAMA [DC98], a hash function attacked in [Rij+02]. It was based on a large "buffer" and a smaller "inner state", the former being linearly updated using message blocks, and the latter being non-linearly updated using data extracted from the buffer. Our construction is presented in the next section.

[3] This is already observed by the designers of Rocca who observed the negative impact that AES and XOR used "in a cascade way" could have.

7.1.2 A family of universal hash functions

We present a family of UHFs, that is large-enough to contain algorithms that are both fast and secure, but also small-enough so that vast subsets of it can be explored in practice.

The idea is to separate the (potentially large) state into two subparts with different roles:

- an *inner part* updated with AES rounds and a linear layer,
- and an *outer part* updated only with a linear layer and new message blocks.

Each round, words of the outer state are XORed to the inner state, but not the other way around. Thanks to the outer part, each message block is XORed several times into the inner state, so that short differential trails leading to collisions do not exist. This construction is similar to many sponge-like constructions, but in our case the linear outer state allows to save many AES round calls. On the other hand, sponge-like designs would apply the same function to the full state. It can also be thought of as a large tweakable block cipher with a large tweak, and a linear tweakey schedule.

By restricting the application of the AES round function to the inner part, such a construction has the potential to offer both a high throughput and a low rate. The sparsity of the round function also makes easier the security analysis based on MILP modeling that is detailed in Section 7.1.4.

In the following, vectors of 128 bits are either named *word*, *block*, *register* or *wire* depending on the context. As in the previous chapter, a diagonal block matrix whose diagonal is made of matrices $A_0, \cdots, A_\ell$ is denoted by $\mathrm{diag}(A_0, \cdots, A_\ell)$. In a block matrix definition, $\star$ denotes an arbitrary block.

7.1.2.a Overall structure

The UHF family we consider is described in Figure 7.1. Each *wire* on the figure represents a 128-bit value. The inner state is represented on the left-hand side, and the outer linear message-schedule with memory on the right. Overall, it looks like a standard an SPN (see Figure 1.4): the inner state, which is alternatively denoted by X, Y and Z, is iteratively updated through a round function built by composing a linear layer with a non-linear one. Between each round, the linear message-schedule ingests several blocks of the input message, and produces an *injected value* V, which is added to Z to yield X. The memory registers of the linear message-schedule, that we denote by R, keep linear information about previous input message blocks.

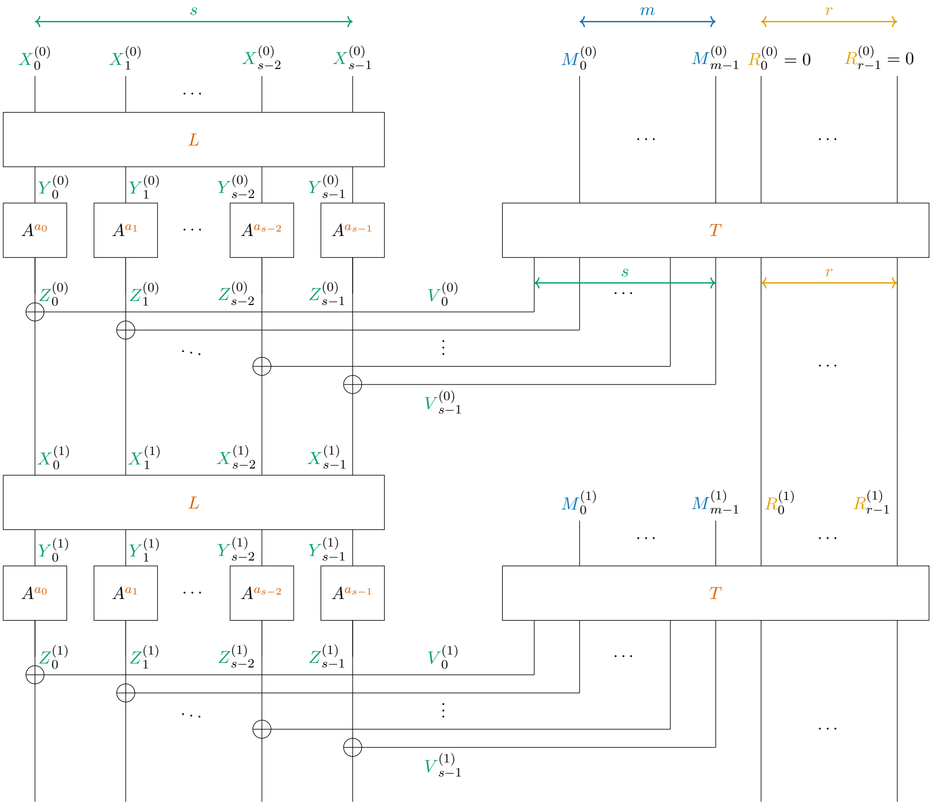


Figure 7.1: A stands for a key-less AES round. Each choice of the size parameters s, m, r, the Boolean values a_i, and the matrices L, T defines an instance.

Parameters. From now on, by *size*, we always mean the number of 128-bit blocks. Thus, each member of the family is parameterized by the sizes s, m, r of the inner state X, Y, Z, of the input message M, and of the memory R, that can be chosen freely. Note that s also corresponds to the size of the injected value V. Once these sizes are fixed, we define a specific instance by choosing the vector a and the matrices L, T.

The Boolean vector $a := (a_0, \cdots, a_{s-1})$, of size s, indicates whether a state wire goes through an AES round or not: $A^0 = \mathrm{Id}$ and A^1 is one round of AES. For any i, if $a_i = 1$, the i-th wire of the state is called an *AES wire*.

The $s \times s$ invertible sparse matrix $L \in \mathbf{GL}_s(\mathbb{F}_{2^{128}})$ is used as linear layer. By design, we restrict the coefficients of L to $\{0, 1\}$, so that L can be viewed as a matrix of $\mathbf{GL}_s(\mathbb{F}_2)$. In particular, the output of the linear layer is only composed of copies and XORs of the 128-bit input words.

Finally, T is the $(s + r) \times (m + r)$ message-schedule *transition matrix*. The matrix T indicates how to compute the s-word injected-value V and how to update the memory R (of size r). Both are linearly computed using the current memory R and m fresh message words, M. Similarly to L, we restrict by design the coefficients of T to $\{0, 1\}$, *i.e.* $T \in \mathbf{M}_{(s+r)\times(m+r)}(\mathbb{F}_2)$.

Notation 7.8 (Time stamp, coordinates and sequences)**.** *We use superscript to indicate the clock (with $t = 0$ as initial clock) and subscripts are reserved for coordinates: for instance $R_i^{(t)}$ stands for the i-th coordinate of $R^{(t)}$, that is, R at time t. We keep plain characters for generic purposes: e.g. the memory R, and use calligraphic letters to denote the sequence throughout time: e.g. $\mathcal{V} := (V^{(t)})_{t\in\mathbb{N}}$. Finally, for any finite subsets $I \subset \mathbb{N}$, $J \subset [\![0, s-1]\!]$ and $t \in \mathbb{N}$, we denote sub-sequences and sub-vectors as: $\mathcal{V}^I := (V^{(t)})_{t\in I}$ and $V_J^{(t)} := (V_j^{(t)})_{j\in J}$.*

7.1.2.b Round function and message-schedule

Round Function. It is applied on the inner state, and is composed of three layers.

Linear layer. The matrix L is applied to X to produce Y: $\forall\, t \geq 0, Y^{(t)} := L\left(X^{(t)}\right)$.

AES-round layer. An AES round A, composed of SubBytes, ShiftRows and MixColumns, but *without* AddRoundKey, is applied in parallel to each AES wire: $\forall\, t \geq 0, i \in [\![0, s-1]\!],\quad Z_i^{(t)} := A^{a_i}\left(Y_i^{(t)}\right)$.

Injected-value addition layer. The injected value V, which is generated by the message-schedule, is added to the state: $\forall\, t \geq 0,\ X^{(t+1)} := Z^{(t)} + V^{(t)}$.

In the AES round layer, the AddRoundKey step is omitted. Thus, by using the AddRoundKey step of the AES-NI instruction, the addition of the round-value word can be considered as free on AES wires.

Message-Schedule. The linear message-schedule has a memory R of size r. Each register contains a linear combination of previous message words. At round t, m new message words are ingested, the s-long injected value $V^{(t)}$ is output and the memory $R^{(t)}$ is updated, in a single transition step:

$$\forall t \geq 0, \quad \begin{pmatrix} R^{(t+1)} \\ V^{(t)} \end{pmatrix} = T \begin{pmatrix} R^{(t)} \\ M^{(t)} \end{pmatrix}. \tag{7.1}$$

As highlighted by the previous equation, it is convenient to decompose T as a block matrix.

Notation 7.9 (T decomposition)**.** *In the following, given a transition matrix T, we use the following decomposition and notation:*

$$T := \begin{array}{cc} \begin{array}{cc} \overset{r}{\longleftrightarrow} & \overset{m}{\longleftrightarrow} \end{array} & \\ \begin{pmatrix} T_{00} & T_{01} \\ T_{10} & T_{11} \end{pmatrix} & \begin{array}{l} \updownarrow r \\ \updownarrow s \end{array} \end{array}. \tag{7.2}$$

Taking advantages of Eqs. (7.1) and (7.2), we can easily express the injected-values as (recursive) linear combinations of input-messages blocks:

$$\forall t \geq 0, \quad R^{(t+1)} = T_{00}R^{(t)} + T_{01}M^{(t)} \qquad V^{(t)} = T_{10}R^{(t)} + T_{11}M^{(t)}. \tag{7.3}$$

Remark 7.10. Let $I \subset \mathbb{N}$, $u := \max(I)$. The sequence $\mathcal{V}^I$ can therefore be viewed as a family of $|I| \cdot s$ linear combinations, or equivalently as a $|I|\, s \times um$ matrix where each column represents one of the um message blocks that can appear in the $|I|\, s$ combinations. We often prefer the latter point of view. ▷

An injected-value sequence $\mathcal{V}$ can be obtained from infinitely many matrices T. For instance, infinitely many unused memory registers could be added. It is thus necessary to limit as much as possible this redundancy while exploring the transition matrices T. In the next section, we start by finding a normal form for the transition matrices. We then limit our search by defining an equivalence relation between injected-value sequences and finally present and justify our search space.

7.1.3 A searchable space of universal hash functions

7.1.3.a A normal form for transition matrices

The first notable point about transition matrices is that, at clock t, only the space spanned by the memory registers (and not the register themselves) matters. Indeed, the same information can be recovered from two different spanning families, only in different representation systems. This is illustrated by the following proposition.

Proposition 7.11 (Change of basis for memory registers)**.** *Let T be a transition matrix. Let $P \in \mathbf{GL}_r(\mathbb{F}_2)$. Let us define $T_P \in \mathbf{M}_{(s+r)\times(m+r)}(\mathbb{F}_2)$ such that:*

$$T_P = \begin{pmatrix} PT_{00}P^{-1} & PT_{01} \\ T_{10}P^{-1} & T_{11} \end{pmatrix}. \tag{7.4}$$

Then T_P produces the same sequence $\mathcal{V}$ as the original matrix T.

Proof. Let us denote for any $t \geq 0$, $R_P^{(t)}, V_P^{(t)}$ the respective memory registers and round-message at clock t produced by T_P. By adapting Eq. (7.3) to T_P, we obtain:

$$\forall t \geq 0 \quad R_P^{(t+1)} = PT_{00}P^{-1}R_P^{(t)} + PT_{01}M^{(t)} \qquad V_P^{(t)} = T_{10}P^{-1}R_P^{(t)} + T_{11}M^{(t)}. \tag{7.5}$$

By design, $R^{(0)} = 0$ and $R_P^{(0)} = 0$ because the memory is initialized as such. In particular $R_P^{(0)} = PR^{(0)}$. Let $t \geq 0$ and let us suppose that $R_P^{(t)} = PR^{(t)}$. Then by injecting $R_P^{(t)} = PR^{(t)}$ into Eq. (7.5) and simplifying we get:

$$\begin{aligned} R_P^{(t+1)} &= PT_{00}P^{-1}R_P^{(t)} + PT_{01}M^{(t)} \\ &= PT_{00}P^{-1}PR^{(t)} + PT_{01}M^{(t)} \\ &= PT_{00}R^{(t)} + PT_{01}M^{(t)} \\ &= P(T_{00}R^{(t)} + T_{01}M^{(t)}) = PR^{(t+1)}. \end{aligned}$$

This first proves by induction that $R_P^{(t)} = PR^{(t)}$ for any $t \geq 0$.

Let $t \geq 0$. According to Eq. (7.5), $V_P^{(t)} = T_{10}P^{-1}R_P^{(t)} + T_{11}M^{(t)}$. Replacing $R_P^{(t)}$ by $PR^{(t)}$, we obtain:

$$V_P^{(t)} = T_{10}P^{-1}PR^{(t)} + T_{11}M^{(t)} = T_{10}R^{(t)} + T_{11}M^{(t)} = V^{(t)};$$

which proves the announced equality for any $t \geq 0$. □

For fixed sizes r, m, s, Proposition 7.11 in particular states that it is sufficient to explore a single representative per similarity class for the top-left block T_{00}. Contrary to the previous chapter, we here use the so-called *Frobenius normal form* which is based on the invariant factors rather than the elementary divisors. In the following, we denote by $\sim_{\text{sim}}$ the similarity equivalence relation between matrices. We recall the following classical result about similarity that can be for instance found in [DF04, Thm. 14, p. 476] or [Gan90, p. 192].

Proposition 7.12 (Frobenius normal form, invariant factors)**.** *Let $M \in \mathbf{M}_n(\mathbb{F}_2)$. Then there exists a unique family $(Q_0, \cdots, Q_{\ell-1})$ of polynomials of $\mathbb{F}_2[X]$ such that:*

$$Q_{\ell-1} \mid \cdots \mid Q_1 \mid Q_0 \quad \textit{and} \quad M \sim_{\text{sim}} \operatorname{Diag}(C_{Q_0}, \cdots, C_{Q_{\ell-1}}).$$

This representative is the Frobenius normal form *of M and the polynomials $(Q_0, \ldots, Q_{\ell-1})$ are the* invariant factors *of M.*

According to Proposition 7.12, it is thus sufficient to exhaust all possible Frobenius normal forms rather than all $r \times r$ matrices for the top left-hand corner. This decreases the search space by a significant factor: for $r = 4$, there are $20160 \approx 2^{14.3}$ matrices in $\mathbf{GL}_4(\mathbb{F}_2)$, but only $14 \approx 2^4$ equivalence classes. Actually, it is easier to count the number of equivalence classes by using the canonical form based on the elementary divisors that is introduced in Proposition 6.30: when $\ell = 4$, there exist 14 possible families of elementary divisors.

On top of that, Proposition 7.11 also allows to get rid of redundant memory registers, as presented in the following corollary.

Corollary 7.13. *Let T be a transition matrix. Let us denote $d = \operatorname{rk}(T_{00}|T_{01})$. Then, there exists an instance using d memory-registers which generates the same sequence $\mathcal{V}$.*

Proof. If $d = r$ then T generates $\mathcal{V}$ and has d memory-registers. Let us now suppose that $d < r$. In that case, we can find $P \in \mathbf{GL}_r(\mathbb{F}_2)$ such that the first $r - d$ rows of $(PT_{00}|PT_{01}) = P(T_{00}|T_{01})$ are all-0. The matrix $(PT_{00}P^{-1}|PT_{01})$ naturally shares the same property, and according to Proposition 7.11, T_P produces the same round-message sequence. But the $r - d$ first empty rows in T_P indicates that the first $r - d$ memory registers will be zero at all time $t \geq 0$, and therefore will never impact the output sequence. T_P can thus be adapted by removing the $r - d$ null rows in the upper half, and removing the corresponding $r - d$ columns in the left-hand half. The obtained matrix T' generates the same sequence with d memory registers. □

Corollary 7.13 states that after choosing a Frobenius normal form for T_{00}, and any value for T_{01}, one can immediately look at the rank of the top half $(T_{00}|T_{01})$. If the top half has not full rank, the study of the matrix comes back to the study of an instance with strictly less memory, that is, a smaller r. If the search is done by increasing values of r, one can only consider a top half with a full rank.

7.1.3.b An equivalence relation for injected-value sequences

Even if we limit redundancies thanks to Proposition 7.11 and Corollary 7.13, for most of values of r, m, s, the associated space of message-schedules remains too big. In particular, it cannot be exhaustively searched, especially if a MILP problem needs to be optimized *for each instance.*

To further reduce the explored space, we first restrict ourselves to matrices T for which $\mathrm{rk}(T_{11}) = m$. Indeed, if $\mathrm{rk}(T_{11}) < m$, only a strict subspace of the messages at round t impacts the injected values at this round. This does not directly generate collisions, since the unused messages can be stored in memory and used in later rounds. However, this requires extra registers whose only purpose is to store the unused injected messages of previous rounds, increasing the memory size r without increasing the security. More precisely, after a few rounds, such an instance behaves as if exactly m message blocks impacted the injected values at each round; the message blocks sequence being slightly slid. So from now on, we restric ourselves to the case $\mathrm{rk}(T_{11}) = m$, and in particular, $s \geq m$.

Secondly, we take into account our adversary in a scenario where it has a full control over the input differences in message blocks (such as a chosen-plaintext scenario). From this point-of-view, the implementation does not matter, only the *actual* decompositions of all $V_i^{(t)}$ as linear combinations of $M_j^{(t')}$ with $i, j \in [\![0, m-1]\!]$ and $t' \leq t$ do. In particular, with n degrees of freedom, such an adversary can choose the differences of n independent $V_i^{(t)}$, rather than just the differences of n message blocks $M_i^{(t)}$. We thus study injected-value sequences up to linear change of variables of the inputs.

Definition 7.14 (Linearly-equivalent injected-values sequences.)**.** Let $\mathcal{V} = (V^{(t)})_{t\in\mathbb{N}}$ and $\mathcal{W} = (W^{(t)})_{t\in\mathbb{N}}$ be sequences of linear combinations such that, for any i, t, $V_i^{(t)}$ depends only on $M_j^{(t')}$, where $j \in [\![0, m-1]\!]$, $t' \leq t \in \mathbb{N}$. Then, $\mathcal{V}$ is *linearly-equivalent* to $\mathcal{W}$ if:

$$\forall\ t \in \mathbb{N} \setminus \{0\}\,,\ \exists\ P^{(t)} \in \mathbf{GL}_{tm}(\mathbb{F}_2), \quad V^{[\![0,t-1]\!]} = W^{[\![0,t-1]\!]} P^{(t)},$$

where $P^{(t)}$ is a *lower triangular block matrix*[4] whose blocks are of size $m \times m$. ▷

Remark 7.15. Let $t \in \mathbb{N} \setminus \{0\}$. The lower triangular form of $P^{(t)}$ implies that the equivalence relation preserves the fact that only variables $M_i^{(t')}$, $t' \leq t$ appear in both $V^{(t)}$ and $W^{(t)}$. ▷

[4] The sequences $V^{[\![0,t-1]\!]}$ and $W^{[\![0,t-1]\!]}$ are here viewed as matrices of dimension $ts \times tm$, see Remark 7.10.

Proposition 7.16. *Linear equivalence of injected-value sequences, as defined in Definition 7.14, is an equivalence relation.*

Proof. The invertible lower triangular matrices is a sub-group of $\mathbf{GL}_{tm}(\mathbb{F}_2)$. Let $t > 0$ and $V^{[\![0,t-1]\!]} = W^{[\![0,t-1]\!]}P^{(t)}$, $W^{[\![0,t-1]\!]} = X^{[\![0,t-1]\!]}Q^{(t)}$. Reflexivity is proved with Id, symmetry using $\left(P^{(t)}\right)^{-1}$ and transitivity using $P^{(t)}Q^{(t)}$. □

Proposition 7.17. *Let T be a transition matrix such that* $\mathrm{rk}(T_{11}) = m$. *Then, up to a wire permutation of the inner state, T produces a sequence $\mathcal{V}$ which is linearly-equivalent to the sequence produced by $\widetilde{T}$, where:*

$$\widetilde{T} = \begin{array}{cc} \begin{array}{cc} \overset{r}{\longleftrightarrow} & \overset{m}{\longleftrightarrow} \end{array} & \\ \left(\begin{array}{cc} \star & \star \\ \hline 0 & \mathrm{Id}_m \\ \star & \star \end{array}\right) & \begin{array}{l} \updownarrow r \\ \updownarrow m \\ \updownarrow s-m \end{array} \end{array} .$$

Proof of Proposition 7.17. By hypothesis, $\mathrm{rk}(T_{11}) = m$, so at each round, the information of the m independent new message blocks is fully contained in m of the round-value blocks. In other words, there exists m indices $I = \{i_0, \cdots, i_{m-1}\}$ such that for any t, $V_I^{(t)} = \begin{pmatrix} F & C \end{pmatrix} \times \mathcal{M}^{[\![0,t]\!]}$, where $C \in \mathbf{GL}_m(\mathbb{F}_2)$ (and $F \in \mathbf{M}_{m\times(t-1)m}(\mathbb{F}_2)$).

Up to a wire permutation, let us assume that $I = [\![0, m-1]\!]$. In that case, $(T_{10}|T_{11})$ can be decomposed, such that C appears in it:

$$\begin{pmatrix} T_{10} & T_{11} \end{pmatrix} = \begin{array}{cc} \begin{array}{cc} \overset{r}{\longleftrightarrow} & \overset{m}{\longleftrightarrow} \end{array} & \\ \begin{pmatrix} B & C \\ D & E \end{pmatrix} & \begin{array}{l} \updownarrow m \\ \updownarrow s-m \end{array} \end{array} . \tag{7.6}$$

Now, let $\ell \in \mathbb{N} \setminus \{0\}$ and let us consider the following change of variables:

$$\forall t \in [\![0, \ell-1]\!], \quad \widetilde{M}^{(t)} := BR^{(t)} + CM^{(t)} \iff C^{-1}(\widetilde{M}^{(t)} - BR^{(t)}) = M^{(t)}.$$

Because $R^{(t)}$ is a linear combination of $M_i^{(t')}$ where $t' < t$, this change of variables corresponds to a lower triangular block matrix $P^{(t)}$ (whose diagonal is only made of C blocks).

Decomposing $V^{(t)}$ as $V^{(t)} = \left(V^{(t)}_{[\![0,m-1]\!]} \mid V^{(t)}_{[\![m,s-1]\!]}\right)$, we can rewrite the linear relations in Eq. (7.3) using the decomposition of $(T_{10}|T_{11})$ given in Eq. (7.6):

$$R^0 = 0, \quad \forall t \geq 0,\ R^{(t+1)} = T_{00}R^{(t)} + T_{01}M^{(t)},$$
$$\forall t \geq 0,\ V^{(t)}_{[\![0,m-1]\!]} = BR^{(t)} + CM^{(t)}, \quad V^{(t)}_{[\![m,s-1]\!]} = DR^{(t)} + EM^{(t)}.$$

By substituting $M^{(t)}$ in the previous equations we obtain $R^0 = 0$ and for any $t \geq 0$:

$$R^{(t+1)} = T_{00}R^{(t)} + T_{01}C^{-1}(\widetilde{M}^{(t)} - BR^{(t)}),$$
$$V^{(t)}_{[\![0,m-1]\!]} = BR^{(t)} + CC^{-1}(\widetilde{M}^{(t)} - BR^{(t)}),$$
$$V^{(t)}_{[\![m,s-1]\!]} = DR^{(t)} + EC^{-1}(\widetilde{M}^{(t)} - BR^{(t)});$$

which, once simplified and reorganized, become:

$$\begin{aligned} R^{(t+1)} &= (T_{00} - T_{01}C^{-1}B)R^{(t)} + T_{01}C^{-1}\widetilde{M}^{(t)},\\ V^{(t)}_{[\![0,m-1]\!]} &= \widetilde{M}^{(t)},\\ V^{(t)}_{[\![m,s-1]\!]} &= (D - EC^{-1}B)R^{(t)} + EC^{-1}\widetilde{M}^{(t)}. \end{aligned}$$

Thus, the sequence $\mathcal{V}$, is linearly-equivalent to the sequence generated by the transition matrix $\widetilde{T}$ defined by:

$$\widetilde{T} := \left(\begin{array}{c|c} T_{00} - T_{01}C^{-1}B & T_{01}C^{-1} \\ \hline 0 & \mathrm{Id}_m \\ D - EC^{-1}B & EC^{-1} \end{array}\right). \tag{7.7}$$

The matrix $\widetilde{T}$ has the announced form. □

We can now present the chosen form for the explored transition matrices.

Theorem 7.18. *Let T be a transition matrix such that $\mathrm{rk}(T_{11}) = m$. Then, up to a wire permutation of the inner state, T produces a sequence $\mathcal{V}$ which is linearly-equivalent to the sequence produced by a matrix $\widetilde{T}$ of the following form:*

$$\widetilde{T} = \begin{array}{cl} \begin{array}{cc} \overset{r}{\longleftrightarrow} & \overset{m}{\longleftrightarrow} \end{array} & \\ \left(\begin{array}{cc} F & \star \\ \hline 0 & \mathrm{Id}_m \\ \star & \star \end{array}\right) & \begin{array}{l} \updownarrow r \\ \updownarrow m \\ \updownarrow s-m \end{array} \end{array}; \tag{7.8}$$

where F is a Frobenius normal form matrix.

Proof. First using Proposition 7.17, we obtain, up to a wire permutation of the inner state, a transition matrix $\widetilde{T}$ of the form given in Eq. (7.7) and which produces a linearly-equivalent sequence. By Proposition 7.11, the top left-hand block can be modified to be in Frobenius normal form. The multiplication of the lower-left part by P^{-1} does not change the fact that the first rows of this block are all-0. The lower-right block is not modified, so Id_m still appears on its first rows. The matrix T' has thus the announced form. □

The class of matrices presented in Theorem 7.18 is not only chosen to make the search more efficient, but also for its *sparsity* in order to guarantee a small implementation cost. Indeed, the Frobenius normal form constitutes a very sparse representative of a similarity class: it is a sparse matrix (a diagonal block matrix) with sparse non-empty blocks (companion blocks). The chosen form for the lower half is also quite sparse with the 0 and Id blocks.

7.1.3.c Constraints on the linear layer

Regarding the linear diffusion matrix L, it should be implementable with a low number of XORs. However, we must ensure that each inner state block at round i will eventually influence all of them. To this end, we use the following metric of diffusion.

Definition 7.19. Let $\hat{L}$ be a matrix identical to a binary matrix L, except that its coefficients are integers. The *diffusion time* of L is the smallest integer i such that all coefficients in $(\hat{L})^i$ are non-zero. If no such integer exists, we set it to $+\infty$. ▷

We consider integers rather than binary field elements so that additions do not cancel out; this is equivalent to considering the iterations of L, such that all XORs in the matrix multiplications are replaced with ORs. Intuitively, this number tells how many rounds are needed to ensure a full diffusion in the inner part, although in some special cases, it is not entirely accurate as there may be some bad interactions between non-AES wires and the linear layer L. In the case where all wires are AES wires, this metric is exactly the number of rounds which guarantee that every output wire depends on every input wire. In our search space, we generate matrices L under weight constraints, often with a weight of $s + 1$ or $s + 2$ so that L can be implemented with 1 or 2 XORs and ignore matrices with high diffusion time: we mostly use a value of around $2 \times s$ in this paper[5].

7.1.3.d The actual explored space

The search method presented above is optimized but heuristic: we stress that we do not assure the minimal sparsity of the studied transition matrices. Still, the explored space contains promising candidates (see Table 7.2), that could be further-optimized later on.

Nevertheless, exhaustive search remains unreachable. Equivalence relations on a and L could be used, but would (and in practice do) interfere with the previous ones. Instead, we restrict the weight of a and L.

7.1.4 Turning collision resistance into a MILP problem

The search space being established, we now focus on assessing the security of the potential UHF candidates, by building an adapted MILP model and then solving it thanks to an optimizer. A MILP model is composed of three objects: *variables*, representing either real numbers or (modular) integers[6], *constraints*, that is, inequalities between $\mathbb{Z}$-affine combinations of variables, and an *objective function* which is a $\mathbb{Z}$-linear combination of variables that needs to be maximized (or minimized) when subjected to the given constraints. A MILP solver, such as Gurobi [Gur23], takes as input a MILP model and returns, if it exists, *values* for the variables that both satisfy the constraints and maximizes (or minimizes) the objective function.

[5] For a rate of 1.75, we show candidates with infinite diffusion time.

[6] "Mixed" in MILP actually highlights the different natures of variables.

7.1.4.a Prior works

The use of MILP modeling for searching differential trails with the highest probability was set to light by Mouha, Wang, Gu & Preneel in 2011 [Mou+11]. Several approaches exist depending on the needed level of precision and the available computational power. In theory, by using one MILP variable for each bit of the state at each round, all the non-linear differential transitions could be modeled (at the cost of *many* constraints). This approach is in practice very costly. For byte-aligned primitives, it is much faster and practical to rather affect a MILP variable to each byte of the state. Yet less precise, such a model enables, if it can be efficiently solved, to determine the minimum number of active Sboxes, from which an upper bound on the probability of the best differential trail can easily be estimated. In the case of AES-based ciphers, this method has become standard, as highlighted by the work on Rocca [Sak+21] or Deoxys-BC [Jea+21]. As in these papers, we consider the byte-wise approach. To do so, we extend Notation 7.8 so that the byte position appears.

Notation 7.20. *The second subscript indicates the byte position: $X^i_{j,\ell}$ is the ℓ-th byte of X^i_j.*

7.1.4.b Our model

In this section we consider a *fixed* candidate: s, m, r and a, T, L are thus fixed. To these constants, we add ρ, the number of rounds of the primitive to model. We first describe our variables and objective.

Variables. Let $i \in [\![0, \rho - 1]\!]$ be a round number, $j \in [\![0, B - 1]\!]$ be a word number, where the bound $B \in \{s, m, r\}$ depends on the register, and $\ell \in [\![0, 15]\!]$ be a byte position. Each byte of the state is modeled as a *binary variable*, that is equal to 0 if the byte is inactive (no difference) and 1 if it is active (non-zero difference). The binary variables $x^i_{j,\ell}, y^i_{j,\ell}, z^i_{j,\ell}, r^i_{j,\ell}, m^i_{j,\ell}, v^i_{j,\ell}$ respectively represents the bytes $X^i_{j,\ell}, Y^i_{j,\ell}, Z^i_{j,\ell}, R^{j,\ell}_i, M^{j,\ell}_i, V^i_{j,\ell}$.

Objective. Our goal is to minimize the number of active Sboxes, represented by the variables $y^i_{j,l}$ on AES wires, *i.e.* $j \in \mathrm{Supp}(a)$. Our objective is then defined by:

$$\mathrm{Obj} := \sum_{i=0}^{\rho-1} \sum_{j \in \mathrm{Supp}(a)} \sum_{\ell=0}^{15} y^i_{j,l}.$$

Before presenting the main constraints, we present auxiliary ones that will appear in the definition of more advanced ones.

Multiple-XOR. It models the relation $\bigoplus_{i=0}^{N-1} U_i = 0$ where $(U_i)_{i\in[\![0,N-1]\!]}$ is a list of N bytes represented by N binary variables $(u_i)_{i\in[\![0,N-1]\!]}$. To do so, we introduce an auxiliary binary variable α, and two constraints:

$$\alpha N \geq \sum_{i=0}^{N-1} u_i \quad \text{and} \quad \sum_{i=0}^{N-1} u_i \geq 2\alpha.$$

In that way, depending on $\alpha \in \{0,1\}$, either 0 or at least 2 bytes are active.

MDS constraints. It models the relation between an input column of bytes, represented by the binary variables $(y_i)_{i\in[\![0,3]\!]}$, and an output one, represented by $(z_i)_{i\in[\![0,3]\!]} \in \{0,1\}^4$, through the AES MDS matrix which a differential branch number equal to 5. This is done using an auxiliary binary variable α, and the two constraints:

$$10\alpha \geq \sum_{i=0}^{3} y_i + z_i \quad \text{and} \quad \sum_{i=0}^{3} y_i + z_i \geq 5\alpha,$$

so that either 0 or at least 5 bytes are active.

Remark 7.21. In the above constraints, Σ corresponds to an integer sum, *not a modulo-2 sum.* ▷

We can now create constraints for each layer of the round function. Let $i \in [\![0, \rho - 1]\!]$.

Linear layer. The transition through L is naturally expressed by linear relations between bytes. If we denote by $L = (L_{j,k})_{j,k\in[\![0,\cdots,s-1]\!]}$, where $L_{j,k} \in \mathbb{F}_2$ for any $j, k,$, it holds that:

$$\forall i \in [\![0, \rho-1]\!], j \in [\![0, s-1]\!], \ell \in [\![0, 15]\!], \qquad Y^i_{j,\ell} = \sum_{k=0}^{s-1} L_{j,k} X^i_{k,\ell}.$$

These constraints can therefore be modeled using a Multiple-XOR constraint.

AES-round layer. Let $j \in \mathrm{Supp}(a)$ so that an AES round is applied on the j-th wire. The Sbox layer does not change the activity pattern, but the linear layer (ShiftRows and MixColumns) needs to be modeled. For any round $i \in [\![0, \rho - 1]\!]$, and column index $t \in [\![0, 3]\!]$, the t-th diagonal of Y^i_j is linked by an MDS relation together with the t-th column Z^i_j. Those relations require an MDS constraint. When $j \notin \mathrm{Supp}(a)$, we simply add the constraints $y^i_{j,\ell} = z^i_{j,\ell}$ for all i, ℓ.

Message-schedule. The 128-bit linear relations between R^i, M^i, R^{i+1}, V^t given by Eq. (7.3) can be modeled with 16 Multiple-XOR constraints (one for each byte).

Injected-value addition. For all i, j, ℓ, $Y^{i+1}_{j,\ell} = Z^i_{j,\ell} + V^i_{j,\ell}$ is modeled as a Multiple-XOR.

Finally, we add constraints on the inputs/outputs of the UHF, and constraints to take advantage of the inherent symmetries of the AES round function.

Input constraints. At clock $t = 0$, the state and memory are fully inactive. Thus:

$$\forall \ell \in [\![0, 15]\!], j \in [\![0, s-1]\!], j' \in [\![0, r-1]\!] \qquad x^0_{j,\ell} = 0, \quad r^0_{j',\ell} = 0.$$

Message constraints. If a trail with an inactive first round exists, shifting it by 1 round makes it still a valid trail. Moreover, in the AES, any column (resp. row) plays the same role, so any trail can be shifted so that the first difference appears in the byte of index $\ell = 0$. By forcing at least one 0-index first-round-message byte to be active, we facilitate the solving process, without leaving any trail aside. Hence the *symmetry constraint*:

$$\sum_{j=0}^{m-1} m^0_{j,0} \geq 1.$$

This model is our *basic model.* Additional constraints can be added to it such as the following output constraints.

Output constraints. We can force the state to be fully inactive at the end:

$$\forall \ell \in [\![0, 15]\!], j \in [\![0, s-1]\!] \qquad x^\rho_{j,\ell} = 0.$$

This last constraint highly reduces the MILP solution space, and in practice enable faster solving by Gurobi. However, it is a too-strong constraint when ρ is small. Indeed, a differential trail over more rounds but with less active Sboxes cannot be captured by this model. In practice, we iteratively increase ρ to capture more and more trails, until a sufficient number of rounds is reached. In the published paper [Bar+24], we also detail other costly constraints to avoid taking into account trails that cannot be instantiated in real life, and therefore deduce more accurate bounds.

A word on solutions. A solution to this model consists in an activity pattern, which, *if it is instantiable*, minimizes the number of active Sboxes. There is however no *a priori* guarantee that it actually can be instantiated as an actual differential trail. Nevertheless, if it is instantiable, and if all transitions can occur with maximal probability, then the instantiated trail would have a probability of p^N, where N is the number of active Sboxes, and $p = \delta_S 2^{-8}$ is the probability associated to the differential uniformity 8-bit Sbox S of the AES. Thanks to Assumption 7.6, this higher bound on the probability of the best differential trail enables us to estimate the level of security of any candidate, once the solver terminates.

Search strategy. The search stategy that we followed was guided by the solving of partial MILP model which can enables us to discard unsecure candidate, but also by our on-the-fly benchmarking tool. Indeed, if a candidate can quickly be set aside because of this low performances, there is no need to solve a costly model. The detailed search strategy, as well a thorough description of our results can be found in [Bar+24, Section 6]. Table 7.2 presents an overview of the variety of the candidates, as well as their competitive performance.

								Speed (cy/B)	
Rate	w	m	s	r	XOR-cost	Diffusion	Security	16 kB	256 kB
2	8	4	9	4	4	15	26	0.074	0.067
1.75	7	4	10	5	5	∞	23	0.079	0.068
2	6	3	7	4	4	11	25	0.086	0.080
2	4	2	6	4	3	9	24	0.104	0.099
2	2	1	4	3	4	5	23	0.180	0.175
2	1	0.5[1]	1	5	3/1[2]	-	26	0.374	0.371

[1] A message is added every other round.
[2] There is 1 inherent XOR in the transition matrix. Every other rounds, the message accounts for 2 additional XORS.

Table 7.2: Retained candidates for different parameters sets. Speeds were measured on Intel 11th Gen Core i5-1135G7 (Tiger Lake) for different message sizes.

7.1.5 MACs based on UHFs

As the previous sections might suggest, the biggest part of this work was to come up with a framework, and to find some interesting candidates. However the ultimate goal remains to build some concrete MAC instances. We detail here the general strategy that we followed and refer to the published paper [Bar+24] and to our GitHub repository[7] for more information about the actual candidate named LeMac and PetitMac.

To turn our fast universal hash function into a MAC, we use the following strategy:

[7] The implementations of LeMac and PetitMac can be found on GitHub at `https://github.com/AugustinBariant/Implementations_LeMac_PetitMac`.

1. Using one of the round functions we obtained in our search, we get an ε-AU family H taking an arbitrary message as input with a $128 \cdot s$-bit output. The family is indexed by the secret initial state, and we conjecture that it is a 2^{-128}-AU family based on our MILP analysis.
2. We compose H with an ε-AXU family C taking a $128 \cdot s$-bit input with a 128-bit output, and therefore obtain an ε'-AXU family $C \circ H$.
3. We use the EWCDM construction [CS16], with the ε'-AXU family $C \circ H$, and the AES block cipher.

We thus obtain a MAC whose security relies only on the pseudo-random-function security of AES and the ε-AU security of H, the former being a standard assumption, and the latter being a consequence of our MILP-based analysis.

ε-AXU family C. We build the family C using the sum hashing construction from [CW77, Proposition 8]. Given two ε-AXU families $H_1 : A_1 \to \mathbb{F}_2^n$ and $H_2 : A_2 \to \mathbb{F}_2^n$, this construction yields an ε-AXU family $G : A_1 \times A_2 \to \mathbb{F}_2^n$ defined by:

$$G = \left\{ x \mapsto (h^{(1)}(x) \oplus h^{(2)}(x)) : h^{(1)} \in H_1, h^{(2)} \in H_2 \right\}.$$

Concretely, we take the AES block cipher as an ε-AXU family (the ε-AXU security of AES is a consequence of its the security as a PRF), and define the family C by:

$$C = \left\{ C_k \colon (x_0, x_1, \dots, x_{s-1}) \mapsto \bigoplus_{i=0}^{s-1} \mathrm{AES}_{k_i}(x_i) \right\}$$

where each AES is keyed independently. C is a 2^{-128}-AXU family assuming that the AES is a secure PRF, and the composition of the 2^{-128}-AU family H and the 2^{-128}-AXU family C yields a 2^{-127}-AXU family $C \circ H$ using the composition result from [Sti92, Theorem 5.6].

EWCDM. The MAC itself follows the EWCDM construction by Cogliati and Seurin [CS16]. Based on a ε-AXU family H and an encryption scheme $\mathcal{E}$, this construction is instanciated as:

$$\mathrm{EWCDM}[H, \mathcal{E}]_{k_1,k_2,k_3}(x, N) = E_{k_3}(H_{k_1}(x) \oplus E_{k_2}(N) \oplus N).$$

It uses a nonce N to obtain high security, but it still provides security up to $2^{n/2}$ queries if the nonces are repeated (or omitted).

When used with unique nonces, EWCDM was initially proved secure up to $2^{2n/3}$ queries, but a more recent result proved security up to essentially 2^n queries [MN17]. We use the EWCDM construction because it provides significantly higher security than the more common Wegman-Carter-Shoup construction:

$$\mathrm{WCS}[H, \mathcal{E}]_{k_1,k_2}(x, N) = H_{k_1}(x) \oplus E_{k_2}(N).$$

Indeed, Wegman-Carter-Shoup only provides $2^{n/2}$ security with unique nonces, and fails completely when nonces are repeated.

Initialization. While the family is indexed by the secret initial state, we suggest to derive it as follows: the branch with index i is initialized to $E_{\mathsf{Kinit}}(i)$, where Kinit is 128-bit secret key, and E is the AES-128 block cipher.

LeMac & PetitMac. In the end, we present two MACs both of which take as input a 128-bit nonce, a 128-bit key, and return a 128-bit digest. It is based on the round function which corresponds to the fastest candidate we found for $w = 8$. For cases where the high parallel potential of LeMac might not be an advantage (e.g. on smaller processors), we propose instead PetitMac, which is based on the promising candidate we found for $w = 1$. It has a rate of 2, and ensures the activation of at least 26 S-boxes during absorption.

Security & performances. Regarding security, we claim that LeMac and PetitMac offer 128-bit security in the nonce-respecting model, meaning that an attacker with advantage close to one requires a data complexity close to 2^{128}, or a time complexity close to 2^{128}. In the nonce-misuse setting, we claim that an attacker with advantage close to one requires a data complexity close to 2^{64}, or a time complexity close to 2^{128}.

According to our benchmarks that are detailed in [Bar+24, Section 7], LeMac is currently by far the fastest MAC on the high-profile use-case of AES-NI platforms, PetitMac is, as expected, not competitive on high-end desktop but is very competitive on microcontrollers.

7.2 A stream cipher for transciphering in TFHE

We describe in this section our second design, which is a stream cipher operating over $\mathbb{F}_{17}$ to be used for transciphering in TFHE. To do so, we start by presenting the context of fully homomorphic encryption, and deduce our design goals from it.

7.2.1 Context and design goals

7.2.1.a Specificities of TFHE

As presented in introduction of this chapter, we focus now on designing a stream cipher to be used as a transciphering [NLV11] method in TFHE [Chi+16, Chi+17, Chi+20]. Before presenting our design goals, we give more detail about specificities of TFHE which guide our approach, and about the previous ciphers designed to be used as transciphering schemes for TFHE.

In the following, we denote by $\mathbb{Z}_q$ the ring of integers modulo q, where $q \geq 1$. Note that q stands for *quotient* and can *a priori* take any integer value. The *discretized torus* is defined by $\mathbb{T}_q = \{\frac{a}{q} \mid a \in \mathbb{Z}_q\}$.

The security of TFHE is based on the following variants of the Learning With Errors (LWE) problem [Reg05].

Definition 7.22 (LWE problem over the discretized torus)**.** Let $q, n \in \mathbb{N}$ and let $s = (s_1, \ldots, s_n) \xleftarrow{\$} \mathbb{F}_2^n$. Let χ be an error distribution over $\mathbb{Z}_q$. The *decisional learning with errors over discretized torus problem* is to distinguish between samples drawn from the two following distributions:

$$\mathcal{D}_0 = \{(a, r) \mid a \xleftarrow{\$} \mathbb{T}_q^n, r \xleftarrow{\$} \mathbb{T}_q\}$$

$$\mathcal{D}_1 = \{(a, b) \mid a = (a_1, \ldots, a_n) \xleftarrow{\$} \mathbb{T}_q^n, e \xleftarrow{\$} \chi, b = \sum_{j=1}^{n} a_j \cdot s_j + e\}.$$

The *search* version of the problem is to recover s from samples of $\mathcal{D}_1$, and the generalized decisional and search problems [BGV12] are defined similarly, but for a_i, r drawn from $\mathbb{T}_q[X]/(X^n+1)$ and s_i from $\mathbb{F}_2[X]/(X^n+1)$, where in that case n is a power of two. ▷

We do not present the TFHE scheme in detail and refer to [Chi+16, Chi+17, Chi+20] or to our paper [Bau+24a] for more information. However, it is important to define the *plaintext space* in order to understand the specificities of our stream cipher.

Plaintext space. The plaintext space is the ring $\mathbb{Z}_p$, with $p \in \mathbb{N}$. We identify $\mathbb{Z}_p$ with $\mathbb{T}_p$. Furthermore, $\mathbb{Z}_p$ is embedded in $\mathbb{Z}_q$, with $q > p$ using the mapping $\rho : \mathbb{Z}_p \to \mathbb{Z}_q$, defined by $\rho : m \mapsto \left\lfloor \frac{mq}{p} \right\rceil$. The image of this mapping only reaches p elements in $\mathbb{Z}_q$, which take the form $\left\{ \frac{kq}{p} \mid k \in \mathbb{Z}_p \right\}$. These elements are evenly distributed across $\mathbb{Z}_q$ and form *sectors of* $\mathbb{Z}_q$, represented by: $\left\{ \left(\frac{(2k-1)q}{2p}, \frac{(2k+1)q}{2p} \right) \mid k \in \mathbb{Z}_p \right\}$.

During the encryption of m, a small *noise* term e is sampled from a Gaussian distribution over $\mathbb{Z}_q$ and added to m. Since e is small, the noisy message $m + e$ remains within the same sector as m. However, as homomorphic operations are performed, the noise grows and may eventually exceed the sector boundaries. Upon decryption, the recovered message takes the form $m' + e'$, where m' is the expected result and e' is the accumulated noise. As long as $e' < \frac{q}{2p}$, the message m' can be correctly recovered by rounding to the nearest sector center. The noise in TFHE grows depending on the operations.

Sum of ciphertexts. The sum of two ciphertexts c_1 and c_2 that repectively encrypts m_1 and m_2 with noise levels σ_1^2 and σ_2^2, results in a valid ciphertext c', which encrypts $m_1 + m_2$ with noise $\sigma_1^2 + \sigma_2^2$.

Product with a cleartext. Multiplying each coordinate of c_1 by a constant $\lambda \in \mathbb{Z}$ produces a valid ciphertext c', which encrypts $m' = \lambda \cdot m$ with noise $\lambda^2 \cdot \sigma_1^2$.

These linear operations are extremely fast, particularly in comparison to bootstrapping. However, they increase the noise level, which means that only a limited number of such operations can be performed before the correctness of the results is compromised.

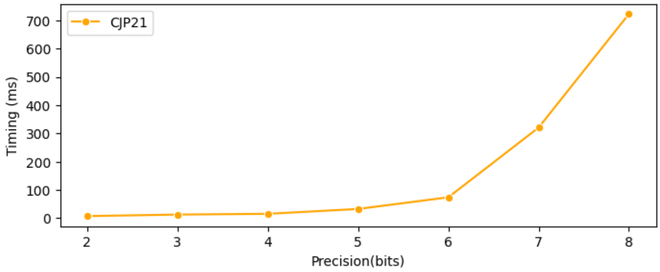


Figure 7.2: Timing of a PBS with respect to the precision of the ciphertext.

Programmable Bootstrapping (PBS). Bootstrapping [Gen09] is a generic technique that allows the noise of a ciphertext to be homomorphically reset to a nominal level. Thanks to this technique, the correctness of long computations can be guaranteed. In TFHE, bootstrapping is impletemented in a *programmable* manner: while the noise is being reset, any arbitrary function can be evaluated on the ciphertext. The programmable bootstrapping (or PBS) is by far the most computationally expensive operation in TFHE, and its cost increases significantly with the modulus p of the plaintext, as illustrated in Figure 7.2.

7.2.1.b State of the art

While transciphering can theoretically be instantiated with any symmetric cipher, traditional ciphers like AES were soon found to be suboptimal [GHS12]. This prompted the exploration of specialized ciphers tailored for transciphering.

Early specialized approaches included the `LowMC` block ciphers [Alb+15] and the `Kreyvium` stream cipher [Can+16]. These ciphers offered reduced multiplicative depths, making them more suitable for homomorphic encryption. Though not initially designed for TFHE, `Trivium` and `Kreyvium` provide good performance within the TFHE transciphering framework [BOS23].

In 2016, the `FLIP` stream cipher [Méa+16] introduced a novel concept based on a filter permutator that randomly permutes key bits and applies a non-linear function to generate a keystream bit. Its key innovation was the direct application of non-linear filtering on key bits, which helped control the noise generated during homomorphic operations. Two variants of `FLIP`, named `FiLIP` [Méa+19] and `Elisabeth` [Cos+22], aimed at a higher security level and improved performance. Most notably, `Elisabeth` operates on arbitrary groups like $\mathbb{Z}_{2^4}$ to minimize costly field conversions in homomorphic evaluations. However, in 2023, an algebraic attack successfully compromised `Elisabeth` [Gil+23]. In response, patched versions—`Elisabeth-b`, `Gabriel`, and `Margrethe`—were proposed [HMS23], but their TFHE evaluation cost was at least double that of the original `Elisabeth` in single-thread computations.

The most recent advancement in transciphering is `FRAST` [Cho+24], which introduces a TFHE-friendly round function based on a random Sbox to reduce the number of rounds. FRAST significantly boosts throughput, though with slightly increased communication overhead and a needed setup phase.

7.2.1.c Our design choices.

In light of this (very light) introduction on TFHE, but mostly thanks to the guidance of coauthors who actively work on FHE, we started designing our stream cipher based on the following guidelines.

Plaintext space. The plaintext space is reduced to a few bits to limit the cost of the PBS. We choose $p = 17$ so that p is odd, as suggested in [BPR24] which avoids having to deal with negacyclicity, and the closest possible to a low power of 2, which is convenient for encoding 4-bit nibbles. Because p is a prime number, it also eases the design and security analysis thanks to the field structure of $\mathbb{Z}_p = \mathbb{F}_p$.

Non-linearity. The non-linearity comes from a layer of Sboxes, each computing a function $\mathbb{F}_p \to \mathbb{F}_p$ giving rise to one PBS evaluation. Given our fixed choice of p, the number of PBS per element of the output stream represents the main performance metric which we search to minimize.

Arrangement of operations. The initial key material (giving rise to fresh TFHE ciphertexts) can go through complex linear combinations before hitting the Sbox layer. Each Sbox output should only go through linear operations with low ℓ_2-norms before undergoing another PBS, so that the noise in input of the PBS is sufficiently low to ensure correctness. It should also go through lightweight linear operations (with low ℓ_2-norms) before being released. This way, the TFHE ciphertexts obtained after the stream-cipher decryption, keep a noise level as close to nominal as possible.

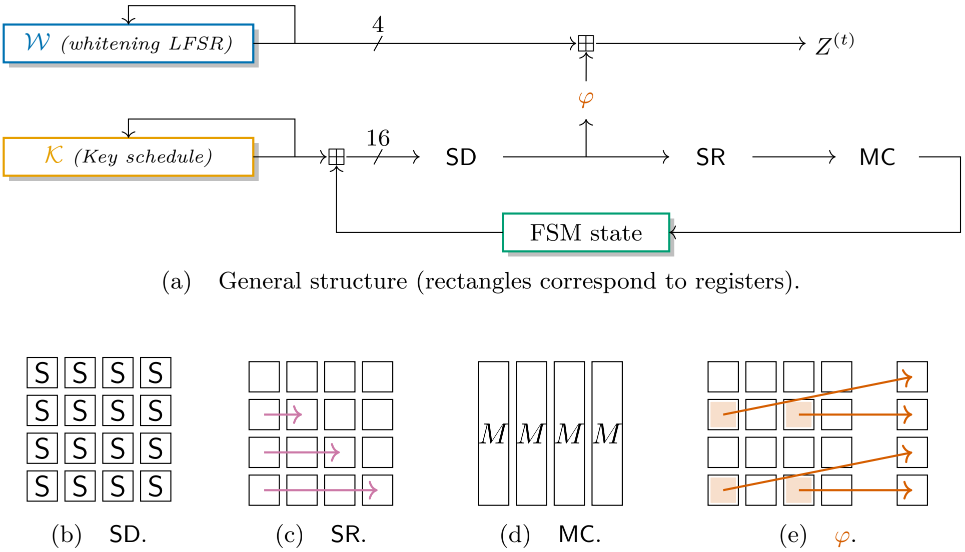
7.2.2 Description of Transistor

Bringing everything together, we designed the stream cipher Transistor. We claim that it provides 128 bits of security, meaning any attack should require at least 2^{128} elementary operations, assuming no more than 2^{31} digits (about 1 GB) are generated with a single master key/IV pair. Transistor is described in this section.

7.2.2.a Overall Structure

Transistor is a stream cipher that generates a keystream composed of elements from $\mathbb{F}_p = \mathbb{F}_{17}$. It generates tuples of 4 digits at once, using the procedure outlined in Figure 7.3.

The idea is to generate two pseudo-random sequences with a very long period using two distincts LFSRs. One of them generates whitening subkeys, while the other acts as a sort of key schedule. The output of the latter is fed into a Finite



(a) General structure (rectangles correspond to registers).

(b) SD.

(c) SR.

(d) MC.

(e) φ.

Figure 7.3: A high level view of `Transistor`.

State Machine (FSM) with its own state, and which operates on it using non-linear operations. We thus have the following components:

- a register of 16 elements of $\mathbb{F}_p$ (the *FSM state*),
- an LFSR over $\mathbb{F}_p$ (the *key schedule* or *key-LFSR* $\mathcal{K}$) of length $|\mathcal{K}| = 64$,
- an LFSR over $\mathbb{F}_p$ (the *whitening LFSR* $\mathcal{W}$), of length $|\mathcal{W}| = 32$,
- a non-linear round function $\mathbb{F}_p^{16} \rightarrow \mathbb{F}_p^{16}$ (the *round function*), and
- a filter $\varphi : \mathbb{F}_p^{16} \rightarrow \mathbb{F}_p^{4}$ that extracts 4 digits from the FSM.

The FSM state is initialized to all zeros, while each LFSR is seeded with key material.

7.2.2.b Detailed Description

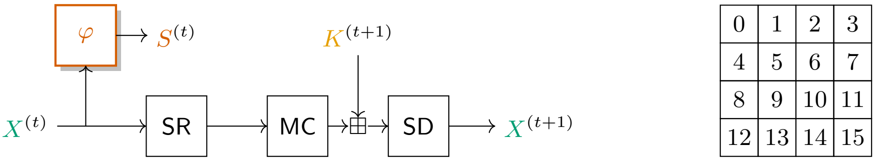
Obviously taking inspiration from the AES, the state of the FSM is organized into a square matrix, where each entry corresponds to a digit in $\mathbb{F}_p$. This matrix is then processed using the following operations:

SubDigits (SD): an S-box layer where a permutation S is applied on each digit;

MixColumns (MC): each column is multiplied by an MDS matrix M over $\mathbb{F}_p$;

ShiftRows (SR): the i-th row is rotated by i positions.

The filter (φ): it extracts 4 digits from the state and returns them.



0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15

(a) Notation throughout clocks. (b) Numbering in the FSM.

Figure 7.4: Our notations.

In what follows, we provide a more detailed description of each step, using the notation summarized in Figure 7.4(a). The keystream output at clock $t \geq 0$ consists of the tuple $Z^{(t)} \in \mathbb{F}_p^4$. The internal state of the FSM, just before the filter is applied, is denoted by $X^{(t)}$, so that $S^{(t)} = \varphi\left(X^{(t)}\right)$. As a consequence, $X^{(t+1)} = \mathsf{SD}\left(K^{(t+1)} + \left(\mathsf{MC} \circ \mathsf{SR}(X^{(t)})\right)\right)$, where $K^{(t)}$ is obtained by concatenating 16 successive digits generated by the key-schedule LFSR $\mathcal{K}$.

S-box Layer (SD). We let S be defined by its lookup table:

$$\mathsf{S} = [\mathtt{1}, \mathtt{12}, \mathtt{6}, \mathtt{11}, \mathtt{14}, \mathtt{3}, \mathtt{15}, \mathtt{5}, \mathtt{10}, \mathtt{9}, \mathtt{13}, \mathtt{16}, \mathtt{7}, \mathtt{8}, \mathtt{0}, \mathtt{2}, \mathtt{4}], \tag{7.9}$$

so that $\mathsf{S}(0) = 1$, $\mathsf{S}(1) = 12$, and so on. It has the following polynomial representation, and is thus of maximum degree:

$$\begin{aligned}\mathsf{S}(x) &= 1 + 4x^1 + 13x^2 + 7x^3 + 16x^4 + 15x^5 + 5x^7 + 5x^8 \\ &\quad + 11x^9 + 13x^{10} + 12x^{11} + 13x^{12} + 15x^{14} + x^{15} \ .\end{aligned}$$

It was chosen by enumerating all APN permutations of $\mathbb{F}_{17}$. Then, we selected S among those that offer a good balance between minimizing the number of pairs $(\Delta^{\text{in}}, \Delta^{\text{out}})$ for which the associated differential equation has exactly two solutions, and minimizing the maximum modulus of the Walsh spectrum.

Linear Layer (MC). We opted for a 4×4 Maximum Distance Separable (MDS) to ensure optimal diffusion. The matrix we chose is:

$$M = \begin{bmatrix} 2 & 1 & 1 & 1 \\ 1 & -1 & 1 & -2 \\ 1 & 1 & -2 & -1 \\ 1 & -2 & -1 & 1 \end{bmatrix} . \tag{7.10}$$

We verified that there is no MDS matrix in $\mathbb{F}_{17}$ with coefficients in $\{-1, 1\}$ by exhaustively testing all such matrices. As we were interested in MDS matrices with minimal ℓ_2-norm and we were able to find during the initial experiments matrices with a squared ℓ_2-norm of 7, it was evident from the definition of the ℓ_2-norm that matrices with minimal ℓ_2-norm could not have coefficients x with $|x| > 2$. Thus, by testing all matrices with coefficients in $\{-2, -1, 1, 2\}$, we found a total of 30 720 MDS matrices with an ℓ_2-norm of 7. We selected M for its symmetries, particularly because it is its own transpose.

Filter. The filter function φ maps the FSM state, which belongs to $\mathbb{F}_p^{16}$, to a tuple (a, b, c, d) in $\mathbb{F}_p^4$. As summarized in Figure 7.3(e), we have that a, b, c and d correspond to the digits of the FSM state with indices 4, 6, 12, and 14 respectively (using the numbering from Figure 7.4(b)).

LFSRs. The whitening LFSR $\mathcal{W}$ and the key schedule LFSR $\mathcal{K}$ are simply LFSRs over $\mathbb{F}_p$ of maximum period, and which are respectively of length 32 and 64. A simple way to obtain a maximum-period LFSR over $\mathbb{F}_p^w$ is to generate a primitive element of $\mathbb{F}_{p^w}$, and then use the coefficients of its minimal polynomial as the taps. The output can then be any non-trivial linear combination of its cells.

Initialization. The key material loaded in the LFSRs is derived from a 128-bit master key, and possibly an IV, using the SHAKE [Sha3] extendable output function. We set the FSM state $X^{(-1)}$ to be all 0, so that the first keystream tuple $S^{(0)}$ is obtained by adding the first outputs of the whitening LFSR with the image under S of some of the first outputs of the key schedule.

7.2.2.c Controlling the evolution of the noise

The design of `Transistor` allows to control the evolution of the noise in the FSM while getting a very low number of PBS per element. An overview of this evolution is depicted in Figure 7.5, but we refer to our paper [Bau+24a] for its thorough analysis. This analysis is based on the careful arrangement of the operations.

We also use an interesting implementation trick for our LFSR, that we called *silent LFSR*. A naive approach for implementing an LFSR homomorphically would be to maintain an encrypted state, and update it by computing a linear combination with the feedback coefficients. This method however causes the noise in the state to accumulate over time, necessitating periodic use of PBS operations to refresh and control the noise growth. However, every output of an LFSR is a linear combination of the cells in its initial state, so by computing on the fly the coefficients of these linear combinations *in clear*, we can evaluate the output of the LFSR at every clock cycle without updating an encrypted version of the internal state. This way, the noise variance in the output of the silent LFSR remains stable over time. This principle is comparable to the approach of `FLIP` [Méa+16] and follow-up works, whereby a key state is queried without being updated.

7.2.3 Security Analysis

In this section we analyze the resistance of `Transistor` against classical attacks and derive lower bounds on the complexity required for these attacks to succeed. The internal parameters, especially the LFSR dimensions, were chosen based on the security analysis results presented here. It has to be noted that many attacks discussed in this section assume that the adversary has access to the sequence $(S^{(t)})_{t\geq 0}$ *prior to its addition with the whitening LFSR*. The corresponding attacks against `Transistor` have therefore a higher complexity since the attacks we describe

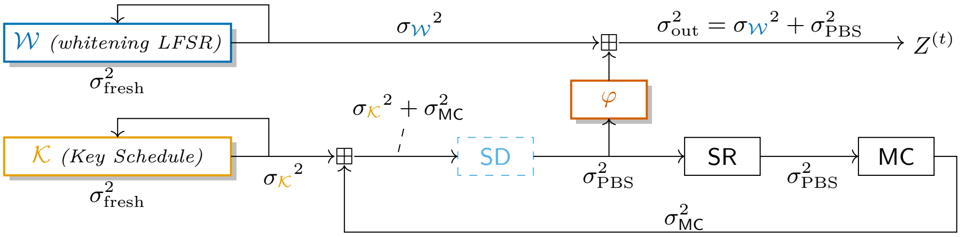

$\mathcal{W}$ (whitening LFSR)
σ^2_{fresh}
$\sigma_{\mathcal{W}}{}^2$
$\sigma^2_{\text{out}} = \sigma_{\mathcal{W}}{}^2 + \sigma^2_{\text{PBS}}$
$Z^{(t)}$
φ
$\sigma_{\mathcal{K}}{}^2 + \sigma^2_{\text{MC}}$
$\mathcal{K}$ (Key Schedule)
σ^2_{fresh}
$\sigma_{\mathcal{K}}{}^2$
SD
σ^2_{PBS}
SR
σ^2_{PBS}
MC
σ^2_{MC}

Figure 7.5: Evolution of the noise variance in a homomorphic evaluation of `Transistor`. Operations involving PBSs are in blue and dashed.

need to be adapted in order to remove the influence of the whitening LFSR. This can be done either by guessing part of its internal state, at the price of a higher time complexity, or by cancelling its outputs thanks to a parity-check equation, at the price of a higher data complexity.

On top of this analysis, algebraic attacks and the relationship between the security of `Transistor` and `LEX` [Bir07] are provided in our paper [Bau+24a].

7.2.3.a Time-Memory-Data trade-offs

Let P, M, T, D denote the respective precomputation, memory, time and data complexities needed for recovering the internal state of a stream cipher. As independently introduced by Babbage [Bab95] and Golic [Gol97], generic Time-Memory-Data Trade-Offs aim at leveraging a more interesting balance between the four metrics than the extreme cases obtained with an exhaustive search ($T = N$, $D = 1$) or a full code-book attack ($P = M = N$, $T = 1$), where N is the number of possible internal states.

To do so, a table is first built and stored offline. This table contains pairs $(X, F(X))$ (indexed by the second coordinate) where F is the function which maps an initial state X to the first n elements of the output sequence where n is chosen such that $\text{Im}(F)$ has size N. Then, during the online phase, the attacker hopes to find a collision between the images stored in the table and the ones observed online. If the attacker observes D sequences of length n, the standard birthday-paradox argument states that $MD \approx N$ is the condition for such a collision to occur. Taking $M = D = P = T = \sqrt{N}$ gives the classical trade-off.

In the case of `Transistor` without the whitening LFSR $\mathcal{W}$, such an attack can be mounted in two ways. First, we can choose F as the function which maps the LFSR state $\mathcal{K}$ *and* the state of the FSM to the first outputs of φ. In this case, $n = |\mathcal{K}| + 16$, providing the first bound:

$$p^{|\mathcal{K}|+16} \geq 2^{\lambda} D \ ,$$

where λ is the security level and $p = 17$ for `Transistor`. In that case, the number of observed sequences D can actually be replaced by the number of observed successive output digits. Indeed, by observing $d \gg n$ output digits, one can build $d - n + 1 \approx d$ sequences of n digits.

Yet, at any clock $t > 0$, the FSM state only depends on the initial state $\mathcal{K}$. We can therefore consider F as the function which maps the initial LFSR state $\mathcal{K}$ (and the all-zero FSM state) to the first $k/4$ output blocks $S^{(0)}, S^{(1)} \dots S^{(k/4-1)}$, where k denotes the size of $\mathcal{K}$ in digits. In this case, $n = |\mathcal{K}|$ and therefore:

$$p^{|\mathcal{K}|} \geq 2^{\lambda} D \ .$$

However, this attack must be launched in the multi-key setting because each observed output sequence must be obtained from an all-zero initial FSM state. Then, D also corresponds to the number of distinct keyed primitives attacked.

Both attacks can be applied to the full `Transistor`, that is, with the whitening LFSR $\mathcal{W}$.

To do so, we denote by $P_{\mathcal{W}} = X^{|\mathcal{W}|} - \sum_{i=1}^{|\mathcal{W}|} c_i X^{|\mathcal{W}|-i}$ the characteristic polynomial of $\mathcal{W}$, and by $(w_t)_{t\in\mathbb{N}}, (s_t)_{t\in\mathbb{N}}, (z_t)_{t\in\mathbb{N}}$ the sequences of digits generated by $\mathcal{W}, \varphi$ and `Transistor` respectively, so that $z_t = w_t + s_t$. Therefore, by linearity, we immediately deduce that,

$$\forall t \geq 0, \quad z_{|\mathcal{W}|+t} - \sum_{i=1}^{|\mathcal{W}|} c_i z_{|\mathcal{W}|-i+t} \ = \ s_{|\mathcal{W}|+t} - \sum_{i=1}^{|\mathcal{W}|} c_i s_{|\mathcal{W}|-i+t}.$$

The same attack as before can therefore be mounted by observing the sequence $(s_{|\mathcal{W}|+t} - \sum_{i=1}^{|\mathcal{W}|} c_i s_{|\mathcal{W}|-i+t})_{t\in\mathbb{N}}$, instead of $(s_t)_{t\in\mathbb{N}}$. Therefore, with the parameters used in `Transistor`, since the length of the keystream generated from the same key is limited to 2^{31} digits, TMDTO attacks have time complexity 2^{296} is the single-key setting, and drops to 2^{130} when keystreams generated from 2^{130} keys are available to the attacker.

7.2.3.b Guess-and-determine attack

We explain a basic guess and determine attack, where the attacker links the FSM state $X^{(t)}$ (initialized as $X^{(-1)} = 0$) to the filter output $S^{(t)}$ by guessing digits of the key-schedule sequence $(K^{(t)})$.

Without the whitening LFSR, the attacker observes at each clock: $t \geq 0$

$$S^{(t)} = \varphi(X^{(t)}) = \mathsf{SD}\left(K^{(t)} + \left(\mathsf{MC} \circ \mathsf{SR}(X^{(t-1)})\right)\right)_I ,$$

where $I = \{4, 6, 12, 14\}$, and where the notation for subvectors is the one introduced in Notation 7.8. Since $X^{(t-1)}$ is known, he deduces:

$$K_I^{(t)} = \mathsf{SD}^{-1}(S^{(t)}) - \left(\mathsf{MC} \circ \mathsf{SR}(X^{(t-1)})\right)_I .$$

After guessing the 12 missing digits $K_J^{(t)}$, where $J = [\![0, 15]\!] \setminus I$, he computes the full $X^{(t)} = \mathsf{SD}\left(K^{(t)} + \left(\mathsf{MC} \circ \mathsf{SR}(X^{(t-1)})\right)\right)$.

Starting from clock $t = |\mathcal{K}|/16$, the key-schedule digits are linearly dependent on the previous ones, and the attacker can verify that the output $S^{(t)}$ is correct without making new guesses. Therefore, in total the adversary has to guess $\frac{12}{16}|\mathcal{K}|$ digits, leading to a complexity $p^{\frac{3}{4}|\mathcal{K}|}$. When taking into account the whitening LFSR, the attacker first has to guess it, leading to an attack with complexity $p^{\frac{3}{4}|\mathcal{K}|+|\mathcal{W}|}$.

7.2.3.c Three consecutive outputs are statistically independent of the secret key

The basic strategy in (fast) correlation attacks against stream ciphers [Sie85, MS88] consists in recovering some information about (a part of) the initial state of the cipher from the knowledge of the keystream. In the following, we investigate this type of attacks *without the whitening LFSR* and aim at recovering the internal state of the key-LFSR $\mathcal{K}$. To this end, we consider the so-called *augmented function* with n outputs, which generates n consecutive output blocks of $\left(S^{(t)}\right)_{t\in\mathbb{N}}$ from the internal state of the FSM and from $(n-1)$ consecutive 16-digit blocks of the key-sequence:

$$\begin{array}{cccc} F^{(n)}: & \mathbb{F}_p^{16}\times\mathbb{F}_p^{16(n-1)} & \to & \mathbb{F}_p^{4n} \\ & \left(X^{(t)},K^{(t+1)},\dots,K^{(t+n-1)}\right) & \mapsto & \left(S^{(t)},S^{(t+1)},\dots,S^{(t+n-1)}\right). \end{array}$$

It is obvious that $F^{(n)}$ is balanced, *i.e.*, all preimages by $F^{(n)}$ have size p^{4n}.

In this context, an important quantity is the smallest length of output sequence $(S^{(t)})_{t\in\mathbb{N}}$ that can provide information on the sequence produced by the key-LFSR [And95]. In the following, we show that this minimal length is at least 4. This property is equivalent to the following theorem.

Theorem 7.23 (Uniform distribution of $F^{(3)}$)**.** *For any $K^{(0)}, K^{(1)} \in \mathbb{F}_p^{16}$, the function*

$$X \mapsto F^{(3)}(X, K^{(0)}, K^{(1)})$$

is uniformly distributed.

The remaining of this section is dedicated to the proof of this theorem. A reader willing to accept this as a fact, can safely continue to Section 7.2.3.d.

Rephrasing the theorem. First, balancedness can be rephrased in terms of differentials by using the following lemma.

Lemma 7.24. *Let $\mathbb{G}$,$\mathbb{K}$ be two groups and let $G : \mathbb{G} \to \mathbb{K}$. Then G is balanced if and only if for all $x \in \mathbb{G}$, it holds that:*

$$|\{\Delta \in \mathbb{G},\ f(x) - f(x+\Delta) = 0\}| \ = \ \frac{|\mathbb{G}|}{|\mathbb{K}|}\,.$$

Proof. Let us denote by y the value $y := f(x)$ and by U_x the set defined by $U_x := \{\Delta \in \mathbb{G}, f(x) - f(x+\Delta) = 0\}$. We immediately observe that:

$$U_x := \{\Delta \in \mathbb{G}, f(x+\Delta) = y\} = \left\{\Delta \in \mathbb{G}, x + \Delta \in G^{-1}(\{y\})\right\} = -x + G^{-1}(\{y\}).$$

In particular, $|U_x| = \left|G^{-1}(\{y\})\right|$, so stating that all preimage sets have cardinality $|\mathbb{G}| \,/\, |\mathbb{K}|$ is equivalent to stating that all sets U_x have cardinality $|\mathbb{G}| \,/\, |\mathbb{K}|$. □

In our case, we consider for any $(K^{(0)}, K^{(0)})$ the function $F : X \mapsto F^{(3)}(X, K^{(0)}, K^{(0)})$. This corresponds to $\mathbb{G} = \mathbb{F}_p^{16}, \mathbb{K} = \mathbb{F}_p^{12}$. It then remains to be shown that for all keys $K^{(0)}, K^{(1)} \in \mathbb{F}_p^{16}$ and for all $X \in \mathbb{F}_p^{16}$ the number of solutions $\Delta \in \mathbb{F}_p^{16}$ of the equation:

$$F^{(3)}(X, K^{(0)}, K^{(1)}) - F^{(3)}(X + \Delta, K^{(0)}, K^{(1)}) = 0$$

is always p^4. This can be done by looking at the number of input differences that lead to 12 output zeros at specific positions. This number can be computed by using that M is an MDS matrix.

MDS code and MDS matrix. While this property was already mentioned in this manuscript, we never developed it. We therefore provide a few classical results.

Definition 7.25 (MDS code and MDS matrix)**.** Let C be an $[n, k, d]$-linear code over $\mathbb{F}_p$, that is, a subspace of dimension k of $\mathbb{F}_p^n$ with a minimal Hamming distance d. The code C is an MDS code if it reaches the Singleton bound [Sin64], *i.e.*, if $d = n - k + 1$. A $t \times t$ matrix M is an MDS matrix if $G = (I_k|M)$ is a generator matrix of an MDS code. ▷

For any matrix G with n columns, and any set $I \subset \{0, \dots, n-1\}$, we denote by G_I the submatrix made of columns whose indices belong to I.

Lemma 7.26 (Characterization of MDS code)**.** *Let C be a linear code of dimension k of $\mathbb{F}_p^n$ and let $G \in \mathbf{M}_{k\times n}(\mathbb{F}_p)$ be a generator matrix of C. Then C is MDS if and only if any $k \times k$ submatrix of G is of full rank (i.e., invertible).*

Proof. Let us assume that there exists I of cardinality k such that G_I satisfies $\operatorname{rank}(G_I) < k$. In that case, there exists a non-zero linear combination $a \in \mathbb{F}_p^k$ of the rows of G_I which sums to 0. Therefore, the codeword $c = aG$ is not zero (because $a \neq 0$ and G of full rank) but c has at least k coordinates which are equal to 0. This implies that $\operatorname{wt}(c) \leq n - k < n - k + 1$ and C is not MDS. Conversely, let us suppose that G_I is of full rank for any I of cardinality k. Let us suppose that there exists $c \in C \setminus \{0\}$ such that $\operatorname{wt}(c) \leq n - k$. Then there exists a set I of k indices such that $c_i = 0$ for any $i \in I$. But because c is not zero, $c = aG$ with $a \neq 0$. However, we also have that $0 = aG_I$. This contradicts the full rank of G_I. □

Lemma 7.27 (Information set). *Let G be the generator matrix of a $[n, k, d]$-code C. Let $I \subset [\![0, n-1]\!]$ be a subset of cardinality k such that G_I is of full rank. Let $J = [\![0, n-1]\!] \setminus I$. Then for any $x \in \mathbb{F}_p^k$ there exists a unique $c \in C$ such that $c_I = x$. In particular, c can be linearly expressed in term of c_I.*

Proof. By construction, $\mathrm{Im}(G_I)$ is the space $\mathrm{Im}(G_I) = \{c_I, c \in C\}$. But because G_I is of full rank, we also have that $\mathrm{Im}(G_I) = \mathbb{F}_p^k$, which proves both existence and uniqueness. Furthermore, let $c \in C$. We observe that $d := (G_I^{-1}(c_I))G$ is by construction a codeword such that $d_I = c_I$. By uniqueness, we conclude that $d = c$ and therefore $c = (G_I^{-1}(c_I))G$. The word c is therefore the image of c_I by the linear mapping $x \mapsto (G_I^{-1}(x))G$. □

We can now state the key proposition for our proof of Theorem 7.23.

Proposition 7.28. *Let M be a 4×4-MDS matrix over $\mathbb{F}_p$. Let $a, b \in \mathbb{F}_p$. Let $W := \mathbb{F}_p \times \{a\} \times \mathbb{F}_p \times \{b\}$. Then there exist two affine functions $L_1, L_2 : (\mathbb{F}_p)^2 \to \mathbb{F}_p$ such that $M(W)$ can be described as:*

$$M(W) = \{(L_1(x, y), x, L_2(x, y), y), x, y \in \mathbb{F}_p\}.$$

Proof. We consider the $[8, 4, 5]$ MDS-code $C = \{(x, Mx), x \in (\mathbb{F}_p)^4\}$. Any set $I \subset \{0, \dots, 7\}$ of cardinality 4 is then an information set. For any $a, b \in \mathbb{F}_p$, we can thus list all codewords where $c_1 = a$ and $c_3 = b$ by letting (c_0, c_2) range over all of $\mathbb{F}_p \times \mathbb{F}_p$. But the previous enumeration can be achieved by fixing $c_1 = a$ and $c_3 = b$, and allowing any pair (c_i, c_j) with $i, j \notin \{1, 3\}$ and $i \neq j$ to take values in $\mathbb{F}_p \times \mathbb{F}_p$, because $\{1, 3, i, j\}$ also forms an information set. We deduce, that when c_1 and c_3 are fixed to a and b respectively and when (c_0, c_2) goes through $\mathbb{F}_p \times \mathbb{F}_p$, then (c_i, c_j) also goes through all $\mathbb{F}_p \times \mathbb{F}_p$. This is the case for instance for $i = 5, j = 7$.

Moreover, since $\{1, 3, 5, 7\}$ is an information set, all others coordinates c_k are linear in c_1, c_3, c_5, c_7. When restricted to the case where $c_1 = a, c_3 = b$, any c_k becomes affine in c_5, and c_7. In particular, this is the case of c_4 and c_6. □

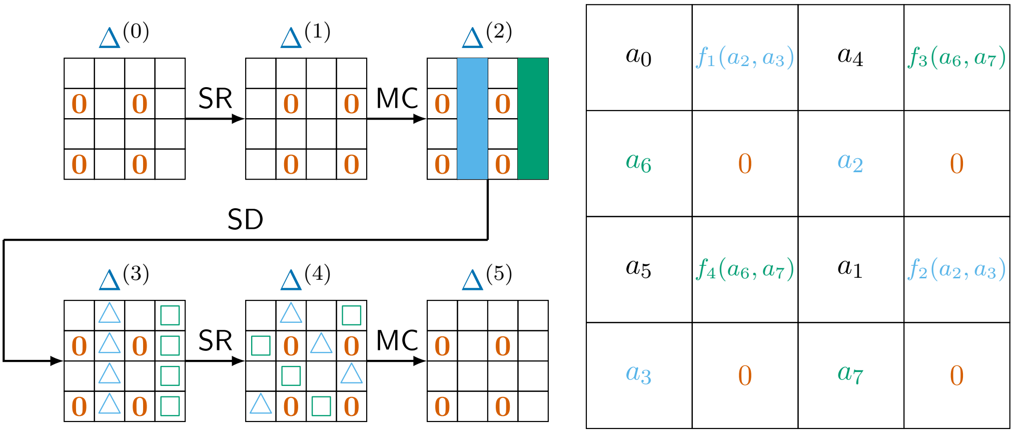
Note that the result can be naturally adapted to any set W of the form $W = U_0 \times U_1 \times U_2 \times U_3$ where two sets U_i are equal to $\mathbb{F}_p$, while the two other are sets with a single element.

Proving Theorem 7.23. We are now ready to prove, which we first reformulate using Lemma 7.24.

Theorem 7.29 (Equivalent formulation of Theorem 7.23). *Let $K^{(0)}, K^{(1)}, X \in \mathbb{F}_p^{16}$ and let us consider the following equation with unknown $\Delta \in \mathbb{F}_p^{16}$:*

$$F^{(3)}(X, K^{(0)}, K^{(1)}) - F^{(3)}(X + \Delta, K^{(0)}, K^{(1)}) = 0. \tag{7.11}$$

Then Eq. (7.11) has p^4 solutions.



(a) Conditions.

a_0	$f_1(a_2,a_3)$	a_4	$f_3(a_6,a_7)$
a_6	0	a_2	0
a_5	$f_4(a_6,a_7)$	a_1	$f_2(a_2,a_3)$
a_3	0	a_7	0

(b) Formal conditions for $\Delta^{(4)}$.

Figure 7.6: The conditions to have $F^{(3)}(X, K^{(0)}, K^{(1)}) - F^{(3)}(X + \Delta, K^{(0)}, K^{(1)}) = 0$.

Proof. Let $K^{(0)}, K^{(1)}, X \in \mathbb{F}_p^{16}$. We look for the number of solutions Δ of the above equation. Let $\Delta^{(i)}$ for $i \in [\![0,5]\!]$ be the possible values of the difference at each relevant step for three consecutive outputs, as shown in Figure 7.6(a). The numbering of the cells can be found in Figure 7.4(b).

Let us first look at the conditions on $\Delta^{(2)}$. First, $\Delta_4^{(2)} = \Delta_6^{(2)} = \Delta_{12}^{(2)} = \Delta_{14}^{(2)} = 0$, since there is no difference in the second output block. Moreover, the second and fourth columns (in blue and green in Figure 7.6(a)) of $\Delta^{(2)}$ must be a vector of the form $M(y, 0, z, 0)$ for a given pair (y, z). Equivalently, let $W := \mathbb{F}_p \times \{0\} \times \mathbb{F}_p \times \{0\}$. Then the second and fourth columns must belong to $M(W)$. Thus, all possible solutions for $\Delta^{(2)}$ that guarantee that the first two output blocks are 0 lie in a vector space V of dimension 8, which is defined by:

$$V := \{(a_0, 0, a_1, 0) || M(a_2, 0, a_3, 0) || (a_4, 0, a_5, 0) || M(a_6, 0, a_7, 0)\} ,$$

where each a_i with $i \in [\![0,7]\!]$ takes all possible values in $\mathbb{F}_p$ and where the concatenation is made column by column. Equivalently, V can be defined by:

$$V = W \times M(W) \times W \times M(W).$$

Moreover, thanks to Proposition 7.28, V can also be defined as the set of words of the form:

$$\begin{gathered}(a_0, 0, a_1, 0) \mid\mid (L_0(a_2, a_3), a_2, L_1(a_2, a_3), a_3) \mid\mid \\ (a_4, 0, a_5, 0) \mid\mid (L_0(a_6, a_7), a_6, L_1(a_6, a_7), a_7),\end{gathered}$$

where each a_i with $i \in [\![0,7]\!]$ takes all possible values in $\mathbb{F}_p$. For each $i \in [\![0,15]\!]$, we denote by $X_i^{(3)}$ and $X_i^{(2)}$ the values of the FSM state at the same instant as $\Delta^{(2)}$ and $\Delta^{(3)}$. Those values are fixed (as we fixed the input and the key). Then the set of values for $\Delta^{(3)}$ such that the first output differences are 0 can be written as $g^X(V)$ where $g^X : \mathbb{F}_p^{16} \to \mathbb{F}_p^{16}$ is the parallel application of some shifted version of the Sbox. Namely, for all $0 \leq i \leq 15$, $g_i^X(b) = \mathsf{S}(X_i^{(2)} + b) - \mathsf{S}(X_i^{(2)}) = S(X_i^{(2)} + b) - X_i^{(3)}$ with $b \in \mathbb{F}_p$. It is worth noting that, since $X^{(2)}$ and $X^{(3)}$ are fixed, all component functions of g^X are bijective because the Sbox is bijective. Also, $g_i^X(0) = 0$. As the function g^X is the concatenation of 16 bijective applications in parallel, then the set of solutions for $\Delta^{(3)}$ can be expressed as $U \times U_X^{(1)} \times U \times U_X^{(2)}$, where $U = \{(a, 0, b, 0), a, b \in \mathbb{F}_p\}$ and $U_X^{(1)}$ can be expressed by:

$$U_X^{(1)} = \left\{ \left(g_1^X \circ \ell_0(a_2, a_3),\ g_5^X(a_2),\ g_9^X \circ \ell_1(a_2, a_3),\ g_{13}^X(a_3) \right), a_2, a_3 \in \mathbb{F}_p^{16} \right\},$$

and $U_X^{(2)}$ can be expressed the same way by using respectively the outputs 3, 7, 11 and 15 of the function g^X. Eventually, as each component of g^X is bijective, the set $U_X^{(1)}$ is of the form:

$$\{(f_1(a_2, a_3),\ a_2,\ f_2(a_2, a_3),\ a_3), a_2, a_3 \in \mathbb{F}_p\},$$

with appropriate functions f_1 and f_2 (and the same holds for $U_X^{(2)}$ for appropriate functions f_3 and f_4. As an example, $f_1(a_2, a_3) := g_1^X \circ L_1((g_5^X)^{-1}(a_2), (g_{13}^X)^{-1}(a_3))$. Eventually, the SR operation is applied to this set and leads to the conditions depicted on Figure 7.6(b). Note that there is no condition on a_0 and a_5. Because of Proposition 7.28, for any fixed value of (a_3, a_6) the image set of the first column $M(\mathbb{F}_p \times \{a_3\} \times \mathbb{F}_p \times \{a_6\})$ contains a single vector of the form $(*, 0, *, 0)$. The same holds for the third column. We then deduce that, for each value of $(a_3, a_6, a_2, a_7) \in \mathbb{F}_p^4$ there is a unique value of $\Delta^{(4)}$ that leads to a zero difference between the third output blocks. We easily check that those solutions indeed satisfy that all output block differences are zero. □

Such a reasoning heavily relies on the specific structure of our FSM, and borrows from the theory of difference propagation for block ciphers.

7.2.3.d Fast correlation attacks involving four consecutive outputs

Now, we want to estimate the minimal data complexity required for recovering the internal state of the key-register from the knowledge of the output sequence $(S^{(t)})_{t \in \mathbb{N}}$, given that at least four consecutive outputs $(S^{(t)}, S^{(t+1)}, S^{(t+2)}, S^{(t+3)})$ need to be considered together. The following lemma shows that, if the output of the augmented function $F^{(n)}$ is correlated to its key-input, then there exists a biased linear relation between the key-inputs and the outputs of $F^{(n)}$. This result is the variant in the non-binary case of the Xiao-Massey lemma [XM88], which can be derived from the proof given by Brynielsson in [Bry89].

Proposition 7.30 (Xiao-Massey lemma over $\mathbb{F}_p$). *Let $F : \mathbb{F}_p^n \times \mathbb{F}_p^m \to \mathbb{F}_p^k$ be a balanced function. Then the function $F_y : \mathbb{F}_p^n \to \mathbb{F}_p^k$ defined by $F_y(X) = F(X, y)$ is balanced for all $y \in \mathbb{F}_p^m$ if and only if the function $(X, Y) \mapsto \alpha \cdot Y + \beta \cdot F(X, Y)$ is balanced for all $\alpha \in \mathbb{F}_p^m$ and all nonzero $\beta \in \mathbb{F}_p^k$.*

It follows that, as soon as the key-LFSR and the considered segment of the output sequence are not statistically independent, there exists a biased linear relation between the digits of these two sequences. There might exist some other relations between these two sequences, of higher degree, whose probability distribution is farther from the uniform distribution. However, it seems much more difficult to exploit nonlinear relations in an attack for two reasons: in practice, what is known to the attacker is the sum between $(S^{(t)})_{t\in\mathbb{N}}$ and the output of the whitening LFSR. Any relation involving the digits of $(S^{(t)})_{t\in\mathbb{N}}$ in a nonlinear manner would involve the outputs of the whitening LFSR is a nonlinear manner, and would then probably require an exhaustive search for its initial state. A second motivation for focusing on linear relations is that recovering the internal state of the key-LFSR faster than exhaustive search is much easier if the known biased relations are linear.

Linear cryptanalysis of $\mathbb{F}_2$. The Fourier transform is the appropriate tool for analyzing the probability distribution of a function. To do so, we overload the notation used for the binary Fourier transform introduced in Proposition 2.13.

Definition 7.31 (Fourier transform over $\mathbb{F}_p$)**.** Let $F : \mathbb{F}_p^n \to \mathbb{F}_p^k$ and ω be a p-th root of unity in $\mathbb{C}$. For $\alpha \in \mathbb{F}_p^n$ and $\beta \in \mathbb{F}_p^k$, the Fourier transform of F is

$$\widehat{F}(\alpha,\beta) = \sum_{x\in\mathbb{F}_p^n} \omega^{\beta\cdot F(x)-\alpha\cdot x}.$$

▷

Note that according to Proposition 2.13, what is defined as $\widehat{F}$ here rather corresponds to the Fourier transforms of the functions $x \mapsto w^{\beta\cdot F(x)}$. However, as observed with Definition 2.15 in the binary case, it is equivalent, but more convenient, to work with the analogous of the Walsh transform. The following proposition provides the minimal length of the sequence $(S^{(t)})_{t\in\mathbb{N}}$ required for recovering the initial state of $(K^{(t)})_{t\in\mathbb{N}}$ using the linear approximation:

$$-\sum_{i=1}^{n-1} \alpha_i \cdot K^{(t+i)} + \sum_{i=0}^{n-1} \beta_i \cdot S^{(t+i)}, \forall t \geq 0\ .$$

This proposition is a particular case of a more general theorem whose proof is detailed in our paper [Bau+24a], and which holds even if the considered approximation is nonlinear, i.e., of the form $-g(K^{(t)},\dots,K^{(t+n-1)}) + h(S^{(t)},\dots,S^{(t+n-1)})$.

Proposition 7.32. *Let $(U_t)_{t\in\mathbb{N}}$ be a sequence of elements in $\mathbb{F}_p^\kappa$ defined by $U_{t+1} = \Phi(U_t)$, $(V_t)_{t\in\mathbb{N}}$ be a sequence of elements in $\mathbb{F}_p^m$ following the uniform distribution and F be a function from $\mathbb{F}_p^\kappa \times \mathbb{F}_p^m$ to $\mathbb{F}_p^n$. Let $\alpha \in \mathbb{F}_p^\kappa$ and $\beta \in \mathbb{F}_p^n$ such that the probability distribution of*

$$(U,V) \mapsto \beta \cdot F(U,V) - \alpha \cdot U$$

is close to the uniform distribution, i.e., for all $z \in \mathbb{F}_p$,

$$\mathbb{P}\left[\beta \cdot F(U,V) - \alpha \cdot U = z\right] = \frac{1}{p} + \varepsilon_z \text{ with } \varepsilon_z \ll 1 \,,$$

where the probability is taken over uniformly random $(U,V) \xleftarrow{\$} \mathbb{F}_p^\kappa \times \mathbb{F}_p^m$. *Let us consider the sequence* $(B^{(t)})_{t\in\mathbb{N}}$ *defined by* $B^{(t)} = \beta \cdot F(U^{(t)}, V^{(t)})$. *Then, the minimal length* N *of* $(B^{(t)})_{t<N}$ *required for recovering* U_0 *is:*

$$N = \frac{\kappa \ln p}{\Delta} \text{ with } \Delta = p \sum_{y\in\mathbb{F}_p} \varepsilon_y^2 = \sum_{b\in\mathbb{F}_p^*} \left| p^{-(\kappa+16)} \widehat{F}(b\alpha, 0; b\beta) \right|^2 \,.$$

It is worth noting that, in the binary case, the previous result corresponds to the usual formula (see e.g. [HN12])

$$\Delta = 4\varepsilon^2 = 2^{-2(\kappa+16)} \left| \sum_{U,V} (-1)^{F(U,V)+\alpha\cdot U} \right|^2 \,.$$

The quantity Δ is called *squared imbalance* of the linear approximation in [BSV07, Def. 4], or the capacity of F [HN12]. Indeed, Δ corresponds to Shannon's capacity of a transmission channel as shown in the proof of the more general theorem in our paper [Bau+24a].

It is worth noting that this bound is tight when U_0 is recovered by performing an exhaustive search over all its possible values and a maximum-likelihood satistical test. This attack has time complexity $Np^{|\mathcal{K}|}$ where $|\mathcal{K}| = 64$. The time complexity decreases if other decoding algorithms are used, for instance algorithms based on low-weight parity-check relations, but the price to pay is a significant increase of the required length N of output sequence [MS88, CT00, CJS01].

Theoretical estimation of the capacity. We now need to estimate the value of the capacity Δ in Proposition 7.32 when F corresponds to the augmented function with n outputs, $F^{(n)}$ for $n \geq 4$. The following theorem proves that the modulus of any Fourier coefficient of the augmented function $F^{(n)}$ is upper bounded by the product of the moduli of Fourier coefficients of all active Sboxes in the corresponding linear trail. Most notably, this implies that the correlation of a linear approximation of several rounds of `Transistor` is determined by a *single linear trail*, and that there is no linear-hull effect as we may find in a block cipher.

Theorem 7.33 (Fourier transform of a linear relation)**.** *Let*

$$\begin{array}{rccl} F^{(n)}: & \mathbb{F}_p^{16} \times \mathbb{F}_p^{16(n-1)} & \to & \mathbb{F}_p^{4n} \\ & (X^{(t)}, K^{(t+1)}, \ldots, K^{(t+n-1)}) & \mapsto & (S^{(t)}, S^{(t+1)}, \ldots, S^{(t+n-1)}) \,. \end{array}$$

Let $\alpha \in \mathbb{F}_p^{16(n-1)}$ *and* β *be any nonzero element in* $\mathbb{F}_p^{4n}$. *Then,*

$$p^{-16n} \left| \widehat{F}^{(n)}(0, \alpha; \beta) \right| \leq \left(\frac{\mathcal{L}(\mathsf{S})}{p} \right)^{w_n}$$

where $\mathcal{L}(\mathsf{S})$ *is the maximal modulus of the Fourier coefficients of* S*, and* $w_n = \sum_{i=1}^{n-1} \mathrm{wt}(\alpha_i)$ *where* $\mathrm{wt}(.)$ *denotes the number of nonzero digits of a vector with coordinates in* $\mathbb{F}_p$*.*

Proof. For any $t \geq 0$, we denote by $X^{(t)}$ the internal state of the FSM at time t after SD. Then we have, as depicted on Figure 7.3(a):

$$X^{(t+1)} = \mathsf{SD}\left(L(X^{(t)}) + K^{(t+1)}\right) \text{ and } S^{(t)} = \varphi(X^{(t)}) ,$$

where L denotes the linear layer, composed of SR and MC. For any $n \geq 1$, we denote by $H^{(n)}$ the following function:

$$\begin{array}{rccc} H^{(n)}: & \mathbb{F}_p^{16} \times \mathbb{F}_p^{16(n-1)} & \to & \mathbb{F}_p^{4(n-1)} \times \mathbb{F}_p^{16} \\ & (X^{(t)}, K^{(t+1)}, \ldots, K^{(t+n-1)}) & \mapsto & (S^{(t)}, \ldots, S^{(t+n-2)}, X^{(t+n-1)}) . \end{array}$$

In other words, the output of $H^{(n)}$ corresponds to the concatenation of the output of $F^{(n)}$ and of the value of the 12-digits of the last state of the FSM which are not outputted by φ. Then, we prove there exists $b' \in \mathbb{F}_p^{16}$ such that

$$\widehat{H}^{(n+1)}(a, \alpha_1, ..., \alpha_n; \beta_0, ..., \beta_{n-1}, b) \;=\; \widehat{\mathsf{SD}}(\alpha_n, b) \times \widehat{H}^{(n)}(a, \alpha_1, ..., \alpha_{n-1}; \beta_0, ..., \beta_{n-2}, b').$$

Let ω be a p-th root of unity in $\mathbb{C}$ and $\chi(x) = \omega^x$ for $x \in \mathbb{F}_p$. For any i, we denote by$\widetilde{K}^{(i)} = \left(K^{(0)}, K^{(1)}, \ldots, K^{(i-1)}\right)$. Then:

$$\begin{aligned} \mathcal{I} &= \widehat{\mathsf{SD}}(\alpha_n, b) \times \widehat{H}^{(n)}(a, \alpha_1, \ldots, \alpha_{n-1}; \beta_0, \ldots, \beta_{n-2}, b') \\ &= \sum_{\widetilde{K}^{(n)}} \chi\left(\sum_{i=0}^{n-2} \beta_i K^{(i)} + b' X^{(n-1)} - \sum_{i=1}^{n-1} \alpha_i K^{(i)} - a K^{(0)}\right) \sum_Z \chi\left(b\mathsf{SD}(Z) - \alpha_n Z\right) \\ &= \sum_{\widetilde{K}^{(n+1)}} \chi\left(\sum_{i=0}^{n-2} \beta_i \cdot K^{(i)} + b' \cdot X^{(n-1)} - \sum_{i=1}^{n-1} \alpha_i \cdot K^{(i)} - a \cdot K^{(0)}\right) \\ &\qquad\qquad \times \chi\left(b \cdot \mathsf{SD}(K^{(n)} + L(X^{(n-1)})) - \alpha_n \cdot K^{(n)} + \alpha_n \cdot L(X^{(n-1)})\right) , \end{aligned}$$

where the variables $K^{(i)}$ are summed over go through $\mathbb{F}_p^{16}$, and the last equality is obtained by setting $Z = K^{(n)} + L(X^{(n-1)})$. We deduce that:

$$\begin{aligned} \mathcal{I} &= \sum_{\widetilde{K}^{(n)}} \chi\left(\sum_{i=0}^{n-1} \beta_i \cdot K^{(i)} - \beta_{n-1} \cdot S^{(n-1)} + b \cdot X^{(n)} - \sum_{i=1}^{n} \alpha_i \cdot K^{(i)} - a \cdot K^{(0)}\right) \\ &\qquad\qquad \times \chi\left(b' \cdot X^{(n-1)} + \alpha_n \cdot L(X^{(n-1)})\right) \\ &= \sum_{\widetilde{K}^{(n)}} \chi\left(\sum_{i=0}^{n-1} \beta_i \cdot K^{(i)} + b \cdot K^{(n)} - \sum_{i=1}^{n} \alpha_i \cdot K^{(i)} - a \cdot K^{(0)}\right) \\ &\qquad\qquad \times \chi\left((b' - \varphi^*(\beta_{n-1}) + L^T(\alpha_n)) \cdot X^{(n-1)}\right) \\ &= \widehat{H}^{(n+1)}(a, \alpha_1, \ldots, \alpha_n; \beta_0, \ldots, \beta_{n-1}, b) , \end{aligned}$$

when $b' = \varphi^*(\beta_{n-1}) - L^T(\alpha_n)$, L^T is the transpose of L and $\varphi^* : \mathbb{F}_p^4 \to \mathbb{F}_p^{16}$ is the function outputting an internal state whose digits are all zero, expect the digits affected by φ, which are equal to the inputs. Moreover, $H^{(1)}$ is the identity function over $\mathbb{F}_p^{16}$ implying that $\widehat{H}^{(1)}(a,b) = p^{16}$ if $a = b$ and 0 otherwise. It follows that there exist $b'_1, \ldots, b'_{n-1}$ such that:

$$\widehat{H}^{(n)}(a, \alpha_1, \ldots \alpha_{n-1}; \beta_0, \ldots, \beta_{n-2}, b) = p^{16} \prod_{i=1}^{n-1} \widehat{\mathsf{SD}}(\alpha_i, b'_i) \,.$$

Therefore,

$$p^{-16n}\widehat{H}^{(n)}(a, \alpha_1, \ldots, \alpha_{n-1}; \beta_0, \ldots, \beta_{n-2}, b) = \prod_{i=1}^{n-1} \frac{\widehat{\mathsf{SD}}(\alpha_i, b'_i)}{p^{16}} \,.$$

The result then directly follows by observing that $\widehat{\mathsf{SD}}(\alpha_i, b'_i)$ is the product of the Fourier coefficients of the 16 Sboxes composing SD. □

Estimation of the capacity in practice. Now, we apply Proposition 7.32 with $U^{(t)} = (K^{(t+1)}, \ldots, K^{(t+n-1)}) \in \mathbb{F}_p^{16(n-1)}$, $V^{(t)} = K^{(t)} \in \mathbb{F}_p^{16}$ and $F(U^{(t)}, V^{(t)}) := F^{(n)}(K^{(t)}, K^{(t+1)}, \ldots, K^{(t+n-1)})$. The previous theorem implies that the data complexity of the best correlation attack based on a linear approximation $\sum_{i=1}^{n-1} \alpha_i \cdot K^{(t+i)} + \sum_{i=0}^{n-1} \beta_i \cdot S^{(t+i)}, \forall t \geq 0$ is the inverse of:

$$\Delta = \frac{p}{64 \ln p} \left(\frac{\mathcal{L}(\mathsf{S})}{p} \right)^{2w_n} \tag{7.12}$$

where $w_n = \sum_{i=1}^{n-1} \mathrm{wt}(\alpha_i)$ is the number of active Sbox in the linear trail.

In order to find a lower bound for the number of S-boxes active in a linear trail over 4 rounds of `Transistor`, we apply a similar approach to the one described in Section 7.1.4. Indeed, it is already mentioned in the seminal work of Mouha, Wang, Gu & Preneel [Mou+11] that, because of the duality between linear and differential cryptanalysis, the same kind of MILP modelization can be used for both kind of attacks. We therefore reuse the already-described MDS constraints, together with some border constraints to ensure that the initial and final inner state are fully inactive, and therefore that the corresponding linear equations do not depend on the unknown FSM state. We also adapt the multiple-XOR constraint to become a so-called *3-fork* constraint.

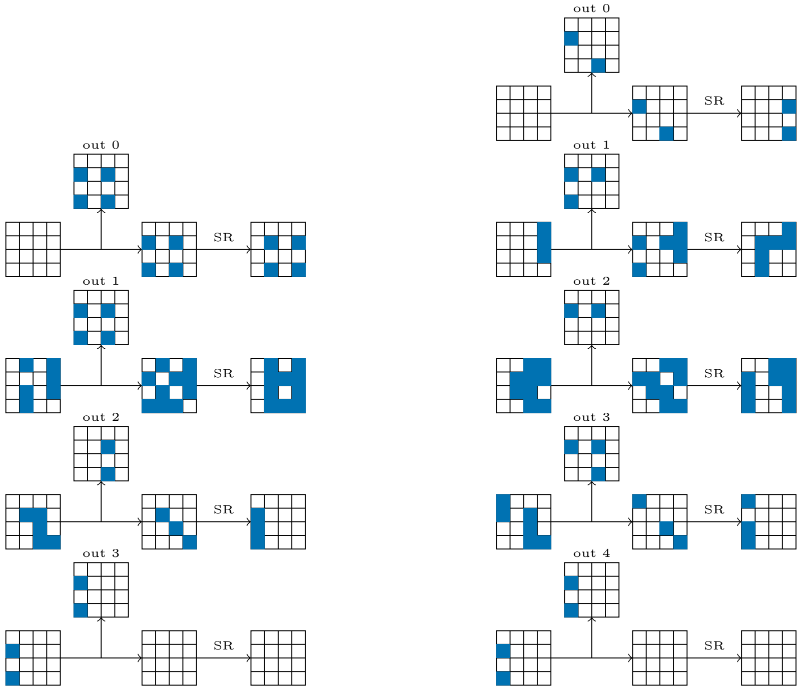
3-fork constraint. In terms of constraints, a sum $X + Y + Z = 0$ (or an equality $X = Y + Z$) can be understood as: "among three such variables if one is active, then at least two of them are". This is exactly the constraint of the multiple-XOR described in Section 7.1.4.

Note that any solution to the associated MILP system is actually a worst-case scenario in our case: a returned activation pattern is not guaranteed to be actually instantiable, just like in the differential case. We solved this simple MILP model using the SageMath interface for Mixed Integer Linear Programing solving within seconds on a standard laptop.

Most notably, we have found that:

$$w_4 \geq 13,\ w_5 \geq 20 \text{ and } w_6 \geq 25$$

with the potential trail examples depicted on Fig. 7.7. We also verified that $w_n \geq 26$ for $n \in \{7, 8, \ldots 26\}$. For larger values of n, either a trail has at least one active Sbox per round, or it splits into two smaller trails with at least 13 active Sboxes. Therefore, we deduce that $w_n \geq 26$ for all $n \geq 7$.



(a) 4-round trail.

(b) 5-round trail.

Figure 7.7: Two activity patterns for linear trails over 4 and 5 rounds.

The Sbox has been chosen to minimize the maximal modulus of its Fourier coefficients, which is $\mathcal{L}(S) = 6.5135$. It follows that the number of 4-digit blocks of $(S^{(t)})_{t\in\mathbb{N}}$ required for any correlation attack based on a linear approximation is at least $2^{39.4}$, corresponding to $2^{41.4}$ digits of the output sequence.

The previous analysis assumes that the output of the whitening LFSR is known by the attacker. This can be achieved by an exhaustive search for its initial state, implying that the time complexity of our attack will be multiplied by a factor $p^{|\mathcal{W}|} \simeq 2^{131}$. However, we can also get rid of the whitening LFSR by choosing some coefficients $\beta_0, \ldots, \beta_{n-1}$ corresponding to a recurring relation satisfied by the sequence generated by $\mathcal{W}$, such as $\sum_{i=0}^{n-1} \beta_i \cdot Z_{t+i} = \sum_{i=0}^{n-1} \beta_i \cdot S^{(t+i)}$. By definition, such a recurring relation corresponds to a multiple of the feedback polynomial of $\mathcal{W}$. It has therefore degree $d \geq 32$, implying that it involves 4-digit blocks of $(S^{(t)})_{t\in\mathbb{N}}$ at distance $\lceil d/4 \rceil \geq 8$. As mentioned, we have $w_n \geq 26$ for $n \geq 7$, which implies that exploiting any linear approximation compatible with a recurring relation for the whitening LFSR requires the knowledge of at least 2^{77} digits of the output sequence.

7.2.4 Performance

Just as in the previous section about LeMac and PetitMac, we refer to our paper [Bau+24a] for a thorough presentation of our implementation and benchmarks. Here, we only compile a few metrics to compare the performances of Transistor with the ones of previous solutions. According to our benchmarks, our design is faster than FiLIP [Méa+19] by several orders of magnitude. But FiLIP was designed to minimize the output noise and not the computation time. Regarding the implementations of Trivium and Kreyvium presented in [BOS23], the performances have been measured on a massive AWS instance. Thus, it is not possible to give a fair comparison with our work. However, we stress that Transistor does not require any set-up time, which is a clear advantage compared to these ciphers.

In Table 7.3, we compare the performances on a similar laptop of Transistor with FRAST [Cho+24], the previous fastest solution in the state of the art. In this paper, the chosen set of parameters is tailored to target a security level of 128 bits and an error probability $p_{\text{err}} = 2^{-80}$. We observe that our instance of Transistor for $p_{\text{err}} = 2^{-128}$ is 3 times faster in terms of throughput with a much lower latency and no setup (against a 25-second setup for FRAST). As FRAST was already faster than Elisabeth and its patches, we did not include them in this comparative study.

Cipher	p_{err}	Setup	Latency	Throughput
FRAST [Cho+24]	2^{-80}	25 s (8 threads)	6.2 s	20.66 bits/s
Transistor	2^{-128}	No	251 ms	65.10 bits/s

Table 7.3: Execution timings of FRAST and Transistor.

7.3 Concluding remarks

Designers of block ciphers can rely on a solid theoretical background that was presented to a great extent in previous chapters. This enables researchers to build secure and efficient primitives. However, for stream ciphers, such frameworks were not really available. With Transistor, we not only outperform the state-of-the-art of stream ciphers tailored for TFHE, we also present a general structure that could easily be adapted to other contexts: the combination of an LFSR-based key schedule, LFSR-based whitening, and non-linear FSM with an AES-like structure is general enough to find applications beyond TFHE.

Our security arguments rely on the minimum number of rounds needed to have a non-zero correlation between the ciphertext and the master key. This is an inherent property of the round function of the FSM which we consider to be of independent interest. A lot of question remains. How does this quantity behave? Can we build FSM update functions in such a way as to increase this quantity without increasing the cost of its evaluation?

For the previously-presented MACs based on the AES, the situation is quite different. Indeed, the relationship between MACs and encryption schemes is better understood, as the large number of AEAD primitives published in the last few years might suggest. However, *dedicated* MACs are not as studied as other constructions. With LeMac and PetitMac, we showcased, again, how much the study of block ciphers can benefit other domains of symmetric cryptography. It also proves once again the influence of the AES on our field, and how much its construction can still heavily influence novel constructions.

Furthermore, we also observe how much automated tools have now a substantial role in cryptanalysis, and can be used for multiple and very different purposes.

While these projects were carried out independently (but with a non-trivial intersection of co-authors), and while the targeted designs have *a priori* little in common, regrouping them in a single chapter enabled the author of this thesis (at least) to identify, in hindsight, a surprising number of similarities between them.

Conclusion and perspectives

The central theme of this manuscript was the following question:

> How can the strengths or weaknesses of symmetric ciphers be captured by their algebraic properties?

This was achieved by multiple approaches. First, in Chapter 3, we presented a higher-order differential attack against the soon-to-be lightweight standard of the American National Institute of Standards and Technology, Ascon. The main properties leveraged in this attack are the sparsity of the polynomial representation of the cipher, but also the low degree of a few iterations of the round function. These specific features enable us to track the evolution of its ANF round after round. While it does not put Ascon at risk when used properly, it surely points out that its lightweightness affects its algebraic representation and allows practical recoveries of information (in misuse scenarios) that are not expected for an ideal cipher.

In Chapters 4 and 5, the lightweightness of the block cipher Midori was challenged by other means. The most striking example is how much this cipher appears secure at first sight, in its original representation, while changing our point of view reveal potential issues that were hidden until that point. More precisely, by using standard and well-studied attacks but in a different system of coordinates, it is possible to discover unexpected distinguishers. By studying a conjugate cipher, rather than the original one, we free ourselves from the initial point of view to rather focus on properties that remain invariant by conjugation, for instance, the cycle decomposition of the function. We also show in these chapters how such considerations are, after all, not so far away from multiple less-standard cryptanalysis techniques. This not only confirms the relevance of the study of conjugate ciphers, it also, and mainly, indicates that such studies could be strengthened and adapted to other targets.

In Chapter 6, we showcased that analyses "up to a change of variables" could also benefit other subfields of symmetric cryptography. In that case, we focused on the study of non-linear components of ciphers and proved that, despite the different point of views taken by many designers of APN functions, most of them could be considered under a single framework. This unifying frame of reference revisits many questions about our understanding of these optimal objects, but could first and foremost enable us to provide new examples to diversify the already known families of APN functions.

Finally in Chapter 7, we demonstrate that such an algebraic point of view is also well-suited to the design of new primitives. With LeMac and PetitMac, we showed for instance that similarity equivalence provides efficient tools to traverse large classes of functions while looking for both secure and efficient universal hash functions. On the other hand, the analysis we made of our stream cipher `Transistor` proves once again the significance of the spectral point of view, and in particular of Fourier analysis, when providing security arguments for a symmetric cipher, even in unusual use cases, including in odd characteristics.

Throughout these examples, we contributed to a better understanding of the algebraic properties of symmetric ciphers, but we above all opened (and perhaps reopened) many questions whose investigation would surely benefit our field. A selection of the most representative open problems that we mentioned throughout this manuscript are recalled below.

Open problems

Chapter 3: Higher-order differential attacks.

Open Problem 7.34 (Data-optimized cube attack). *Can a cube attack against Ascon take advantage, not only of the coefficient α_u of a monomial of highest degree X^u, but also of other coefficients α_v, where $v \in \mathrm{Prec}(u)$?*

Open Problem 7.35 (Heuristic conditional distinguishers). *Let α_u be a coefficient whose polynomial expression (in key variables) is known, and stored. Can we find an efficient heuristic method that helps to determine conditions under which α_u is biased?*

Chapter 4: Cryptanalysis of a conjugate ciphers.

Open Problem 7.36 (New targets). *Can the cryptanalysis of conjugate ciphers be extended to other targets? To other kind of attacks?*

Open Problem 7.37 (Security analysis). *Are we able to provide some security arguments for all conjugates of a cryptographic function, for instance by leveraging its cycle decomposition? What would be the corresponding design criteria for the Sbox ?*

Chapter 5: Commutative cryptanalysis.

Open Problem 7.38 (Commutative cryptanalysis outside existing frameworks). *Can (probabilistic) affine commutants, which are neither linear nor fixed-point-free involutions, be found and leveraged in an attack against a symmetric cipher?*

Open Problem 7.39 (Differential behavior of existing commutative distinguishers). *Can we provide an in-depth analysis of the differential properties relative to the exhibited commutative distinguishers? See Section 5.5.2.*

Chapter 6: Linearly self-equivalence and APN functions.

Open Problem 7.40 (Linearly self-equivalent representatives of APN functions, Problem 6.47)**.** *Does the CCZ-equivalence class of any APN function contain a linearly self-equivalent mapping?*

Open Problem 7.41 ((Non-quadratic) APN ℓ-variate projective mappings)**.** *Can we build more cyclotomic or ℓ-variate projective mappings? Can we build non-quadratic ones, based on APN monomials other than the Gold power mapping?*

Chapter 7: Design of new primitives.

Open Problem 7.42 (`Transistor`-like stream ciphers)**.** *Can we build a `Transistor`-like stream cipher, with an FSM update function which guarantees a zero correlation between the ciphertexts and the master key for more than 3 rounds, without decreasing the performance?*

Bibliography

[Aes] *Advanced Encryption Standard (AES)*. National Institute of Standards and Technology, NIST FIPS PUB 197, U.S. Department of Commerce. Nov. 2001 (cit. on pp. 2, 4, 359).

[AK19] Ralph Ankele and Stefan Kölbl. "Mind the Gap - A Closer Look at the Security of Block Ciphers against Differential Cryptanalysis". In: *SAC 2018*. Ed. by Carlos Cid and Michael J. Jacobson Jr: vol. 11349. LNCS. Calgary, AB, Canada: Springer, Cham, Switzerland, 2019, pp. 163–190 (cit. on p. 44).

[AL16] Tomer Ashur and Yunwen Liu. "Rotational Cryptanalysis in the Presence of Constants". In: *IACR Transactions on Symmetric Cryptology* 2016.1 (2016), pp. 57–70. ISSN: 2519-173X (cit. on p. 159).

[Alb+15] Martin R. Albrecht, Christian Rechberger, Thomas Schneider, Tyge Tiessen, and Michael Zohner. "Ciphers for MPC and FHE". In: *EUROCRYPT 2015, Part I*. Ed. by Elisabeth Oswald and Marc Fischlin. Vol. 9056. LNCS. Sofia, Bulgaria: Springer, Berlin, Heidelberg, Germany, 2015, pp. 430–454 (cit. on p. 282).

[Alb+16] Martin R. Albrecht, Lorenzo Grassi, Christian Rechberger, Arnab Roy, and Tyge Tiessen. "MiMC: Efficient Encryption and Cryptographic Hashing with Minimal Multiplicative Complexity". In: *ASIACRYPT 2016, Part I*. Ed. by Jung Hee Cheon and Tsuyoshi Takagi. Vol. 10031. LNCS. Hanoi, Vietnam: Springer, Berlin, Heidelberg, Germany, 2016, pp. 191–219 (cit. on p. 35).

[And95] Ross J. Anderson. "Searching for the Optimum Correlation Attack". In: *FSE'94*. Ed. by Bart Preneel. Vol. 1008. LNCS. Leuven, Belgium: Springer, Berlin, Heidelberg, Germany, 1995, pp. 137–143 (cit. on p. 289).

[Aum+09] Jean-Philippe Aumasson, Itai Dinur, Willi Meier, and Adi Shamir. "Cube Testers and Key Recovery Attacks on Reduced-Round MD6 and Trivium". In: *FSE 2009*. Ed. by Orr Dunkelman. Vol. 5665. LNCS. Leuven, Belgium: Springer, Berlin, Heidelberg, Germany, 2009, pp. 1–22 (cit. on pp. 63, 74).

[Aum+10] Jean-Philippe Aumasson, Emilia Käsper, Lars R. Knudsen, Krystian Matusiewicz, Rune Steinsmo Ødegård, Thomas Peyrin, and Martin Schläffer. "Distinguishers for the Compression Function and Output Transformation of Hamsi-256". In: *ACISP 10*. Ed. by Ron Steinfeld and Philip Hawkes. Vol. 6168. LNCS. Sydney, NSW, Australia: Springer, Berlin, Heidelberg, Germany, 2010, pp. 87–103 (cit. on p. 79).

[Ava+23] Roberto Avanzi, Subhadeep Banik, Orr Dunkelman, Maria Eichlseder, Shibam Ghosh, Marcel Nageler, and Francesco Regazzoni. "The QARMAv2 Family of Tweakable Block Ciphers". In: *IACR Transactions on Symmetric Cryptology* 2023.3 (2023), pp. 25–73 (cit. on pp. 109, 360).

[AW07] Amir Akbary and Qiang Wang. "On Polynomials of the Form $x^r f(x^{(q-1)/l})$". In: *International Journal of Mathematics and Mathematical Sciences* 2007 (2007) (cit. on p. 198).

[Bab95] Steve Babbage. *A Space/Time Tradeoff in Exhaustive Search Attacks on Stream Ciphers*. European Convention on Security and Detection, IEE Conference Publication No. 408. 1995 (cit. on p. 287).

[Ban+15] Subhadeep Banik, Andrey Bogdanov, Takanori Isobe, Kyoji Shibutani, Harunaga Hiwatari, Toru Akishita, and Francesco Regazzoni. "Midori: A Block Cipher for Low Energy". In: *ASIACRYPT 2015, Part II*. Ed. by Tetsu Iwata and Jung Hee Cheon. Vol. 9453. LNCS. Auckland, New Zealand: Springer, Berlin, Heidelberg, Germany, 2015, pp. 411–436 (cit. on pp. 109, 111, 115, 116, 123, 184, 360, 363).

[Bar24] Augustin Bariant. "Analysis of AES-based and arithmetization-oriented symmetric cryptography primitives". PhD thesis. Sorbonne Université, June 2024 (cit. on p. 70).

[Bar+24] Augustin Bariant, Jules Baudrin, Gaëtan Leurent, Clara Pernot, Léo Perrin, and Thomas Peyrin. "Fast AES-Based Universal Hash Functions and MACs: featuring LeMac and PetitMac". In: *IACR Transactions on Symmetric Cryptology* 2024.2 (2024), 35–67 (cit. on pp. 258, 264, 277, 278, 280, 367).

[Bau+23] Jules Baudrin, Patrick Felke, Gregor Leander, Patrick Neumann, Léo Perrin, and Lukas Stennes. "Commutative Cryptanalysis Made Practical". In: *IACR Transactions on Symmetric Cryptology* 2023.4 (2023), pp. 299–329 (cit. on pp. 110, 135, 142, 156, 166, 169, 185, 364).

[Bau+24a] Jules Baudrin, Sonia Belaïd, Nicolas Bon, Christina Boura, Anne Canteaut, Gaëtan Leurent, Pascal Paillier, Léo Perrin, Matthieu Rivain, Yann Rotella, and Samuel Tap. "Transitor: a TFHE-friendly Stream Cipher". Under submission. 2024 (cit. on pp. 281, 286, 287, 294, 295, 299).

[Bau+24b] Jules Baudrin, Aurélien Boeuf, Xavier Bonnetain, Alain Couvreur, Mathias Joly, and Léo Perrin. *SboxU*. `https://github.com/lpp-crypto/sboxU`. 2024 (cit. on pp. 166, 167, 170, 171, 254).

[BB02] Elad Barkan and Eli Biham. "In How Many Ways Can You Write Rijndael?" In: *ASIACRYPT 2002*. Ed. by Yuliang Zheng. Vol. 2501. LNCS. Queenstown, New Zealand: Springer, Berlin, Heidelberg, Germany, 2002, pp. 160–175 (cit. on p. 160).

[BBL21] Christof Beierle, Marcus Brinkmann, and Gregor Leander. "Linearly Self-Equivalent APN Permutations in Small Dimension". In: *IEEE Transactions on Information Theory* 67.7 (2021), pp. 4863–4875 (cit. on pp. 166, 190, 191, 201, 202, 206, 208, 211, 214, 217, 218, 220).

[BC08] Lilya Budaghyan and Claude Carlet. "Classes of Quadratic APN Trinomials and Hexanomials and Related Structures". In: *IEEE Transactions on Information Theory* 54.5 (2008), pp. 2354–2357 (cit. on pp. 215, 217, 346).

[BC11] Christina Boura and Anne Canteaut. "Zero-Sum Distinguishers for Iterated Permutations and Application to Keccak-f and Hamsi-256". In: *SAC 2010*. Ed. by Alex Biryukov, Guang Gong, and Douglas R. Stinson. Vol. 6544. LNCS. Waterloo, Ontario, Canada: Springer, Berlin, Heidelberg, Germany, 2011, pp. 1–17 (cit. on pp. 32, 79).

[BC13] Christina Boura and Anne Canteaut. "On the Influence of the Algebraic Degree of F^{-1} on the Algebraic Degree of $G \circ F$". In: *IEEE Transactions on Information Theory* 59.1 (2013), pp. 691–702 (cit. on p. 32).

[BC16] Christina Boura and Anne Canteaut. "Another View of the Division Property". In: *CRYPTO 2016, Part I*. Ed. by Matthew Robshaw and Jonathan Katz. Vol. 9814. LNCS. Santa Barbara, CA, USA: Springer, Berlin, Heidelberg, Germany, 2016, pp. 654–682 (cit. on pp. 71, 72).

[BCC19] Christina Boura, Anne Canteaut, and Daniel Coggia. "A General Proof Framework for Recent AES Distinguishers". In: *IACR Transactions on Symmetric Cryptology* 2019.1 (2019), pp. 170–191. ISSN: 2519-173X (cit. on p. 109).

[BCD11] Christina Boura, Anne Canteaut, and Christophe De Cannière. "Higher-Order Differential Properties of Keccak and Luffa". In: *FSE 2011*. Ed. by Antoine Joux. Vol. 6733. LNCS. Lyngby, Denmark: Springer, Berlin, Heidelberg, Germany, 2011, pp. 252–269 (cit. on p. 32).

[BCL06] Lilya Budaghyan, Claude Carlet, and Gregor Leander. *Another class of quadratic APN binomials over $\mathbb{F}_{2^n}$: the case n divisible by 4*. Cryptology ePrint Archive, Report 2006/428. 2006 (cit. on pp. 215, 346).

[BCL08] Lilya Budaghyan, Claude Carlet, and Gregor Leander. “Two Classes of Quadratic APN Binomials Inequivalent to Power Functions”. In: *IEEE Transactions on Information Theory* 54.9 (2008), pp. 4218–4229 (cit. on pp. 215, 343, 346).

[BCL09a] Lilya Budaghyan, Claude Carlet, and Gregor Leander. “Constructing new APN functions from known ones”. In: *Finite Fields and Their Applications* 15.2 (2009), pp. 150–159. ISSN: 1071-5797 (cit. on pp. 215, 343, 346).

[BCL09b] Lilya Budaghyan, Claude Carlet, and Gregor Leander. “On a construction of quadratic APN functions”. In: *2009 IEEE Information Theory Workshop*. 2009, pp. 374–378 (cit. on pp. 215, 220, 222, 254, 343, 346).

[BCL18] Christof Beierle, Anne Canteaut, and Gregor Leander. “Nonlinear Approximations in Cryptanalysis Revisited”. In: *IACR Transactions on Symmetric Cryptology* 2018.4 (2018), pp. 80–101. ISSN: 2519-173X (cit. on pp. 109, 124–128, 134, 137, 153, 187, 363, 364).

[BCP06] Lilya Budaghyan, Claude Carlet, and Alexander Pott. “New classes of almost bent and almost perfect nonlinear polynomials”. In: *IEEE Transactions on Information Theory* 52.3 (2006), pp. 1141–1152 (cit. on pp. 57, 189, 190).

[BCP22] Jules Baudrin, Anne Canteaut, and Léo Perrin. “Practical Cube Attack against Nonce-Misused Ascon”. In: *IACR Transactions on Symmetric Cryptology* 2022.4 (2022), pp. 120–144 (cit. on pp. 64, 82, 362).

[BCP23] Clémence Bouvier, Anne Canteaut, and Léo Perrin. “On the algebraic degree of iterated power functions”. In: *Designs, Codes and Cryptography* 91.3 (2023), pp. 997–1033 (cit. on p. 35).

[BCP24] Jules Baudrin, Anne Canteaut, and Léo Perrin. “On functions of $\mathbb{F}_{2^{2t}}$ mapping cosets of $\mathbb{F}_{2^t}^*$ to cosets of $\mathbb{F}_{2^t}^*$”. In: *WCC 2024: The Thirteenth International Workshop on Coding and Cryptography*. https://wcc2024.sites.dmi.unipg.it/. Perugia, Italy, 2024, pp. 45–57 (cit. on pp. 192, 365).

[BCV20] Lilya Budaghyan, Marco Calderini, and Irene Villa. “On equivalence between known families of quadratic APN functions”. In: *Finite Fields and Their Applications* 66 (2020), p. 101704. ISSN: 1071-5797 (cit. on pp. 215, 217, 343, 346).

[BD08] Eli Biham and Orr Dunkelman. *The SHAvite-3 Hash Function*. Submission to NIST SHA-3 Cryptographic Hash Algorithm Competition. Available at https://www.cs.rit.edu/~ark/20090927/Round2Candidates/SHAvite-3.pdf. 2008 (cit. on p. 260).

[BD94] Thomas Beth and Cunsheng Ding. “On Almost Perfect Nonlinear Permutations”. In: *EUROCRYPT’93*. Ed. by Tor Helleseth. Vol. 765. LNCS. Lofthus, Norway: Springer, Berlin, Heidelberg, Germany, 1994, pp. 65–76 (cit. on p. 189).

[Bea+13] Ray Beaulieu, Douglas Shors, Jason Smith, Stefan Treatman-Clark, Bryan Weeks, and Louis Wingers. *The SIMON and SPECK Families of Lightweight Block Ciphers*. Cryptology ePrint Archive, Report 2013/404. 2013 (cit. on p. 10).

[Bei+16] Christof Beierle, Jérémy Jean, Stefan Kölbl, Gregor Leander, Amir Moradi, Thomas Peyrin, Yu Sasaki, Pascal Sasdrich, and Siang Meng Sim. “The SKINNY Family of Block Ciphers and Its Low-Latency Variant MANTIS”. In: *CRYPTO 2016, Part II*. Ed. by Matthew Robshaw and Jonathan Katz. Vol. 9815. LNCS. Santa Barbara, CA, USA: Springer, Berlin, Heidelberg, Germany, 2016, pp. 123–153 (cit. on pp. 109, 111, 360).

[Bei+22] Christof Beierle, Tim Beyne, Patrick Felke, and Gregor Leander. “Constructing and Deconstructing Intentional Weaknesses in Symmetric Ciphers”. In: *CRYPTO 2022, Part III*. Ed. by Yevgeniy Dodis and Thomas Shrimpton. Vol. 13509. LNCS. Santa Barbara, CA, USA: Springer, Cham, Switzerland, 2022, pp. 748–778 (cit. on p. 162).

[Bei+23] Christof Beierle, Patrick Felke, Gregor Leander, Patrick P. Neumann, and Lukas Stennes. “On Perfect Linear Approximations and Differentials over Two-Round SPNs”. In: *CRYPTO 2023, Part III*. Ed. by Helena Handschuh and Anna Lysyanskaya. Vol. 14083. LNCS. Santa Barbara, CA, USA: Springer, Cham, Switzerland, 2023, pp. 209–239 (cit. on p. 162).

[Ben+08] Ryad Benadjila, Olivier Billet, Henri Gilbert, Gilles Macario-Rat, Thomas Peyrin, Matt Robshaw, and Yannick Seurin. *SHA-3 Proposal: ECHO*. Submission to NIST SHA-3 Cryptographic Hash Algorithm Competition. Available at https://ehash.iaik.tugraz.at/uploads/9/91/Echo.pdf. 2008 (cit. on p. 260).

[Ber05] Daniel J. Bernstein. “The Poly1305-AES Message-Authentication Code”. In: *FSE 2005*. Ed. by Henri Gilbert and Helena Handschuh. Vol. 3557. LNCS. Paris, France: Springer, Berlin, Heidelberg, Germany, 2005, pp. 32–49 (cit. on p. 262).

[Ber+06] Thierry P Berger, Anne Canteaut, Pascale Charpin, and Yann Laigle-Chapuy. “On Almost Perfect Nonlinear Functions Over $\mathbb{F}_2^n$”. In: *IEEE Transactions on Information Theory* 52.9 (2006), pp. 4160–4170 (cit. on p. 237).

[Ber+07] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. “Sponge Functions”. In: *Ecrypt Hash Workshop*. https://keccak.team/files/SpongeFunctions.pdf. 2007 (cit. on pp. 75, 76, 362).

[Ber+11] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. *The Keccak reference*. Tech. rep. https://keccak.team/files/Keccak-reference-3.0.pdf. 2011 (cit. on pp. 62, 78, 102).

[Ber+12a] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. "Duplexing the Sponge: Single-Pass Authenticated Encryption and Other Applications". In: *SAC 2011*. Ed. by Ali Miri and Serge Vaudenay. Vol. 7118. LNCS. Toronto, Ontario, Canada: Springer, Berlin, Heidelberg, Germany, 2012, pp. 320–337 (cit. on pp. 76, 85).

[Ber+12b] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. "Permutation-based encryption, authentication and authenticated encryption". In: *Directions in Authenticated Ciphers*. https://keccak.team/files/KeccakDIAC2012.pdf. 2012 (cit. on p. 85).

[Bey18] Tim Beyne. "Block Cipher Invariants as Eigenvectors of Correlation Matrices". In: *ASIACRYPT 2018, Part I*. Ed. by Thomas Peyrin and Steven Galbraith. Vol. 11272. LNCS. Brisbane, Queensland, Australia: Springer, Cham, Switzerland, 2018, pp. 3–31 (cit. on pp. 109, 115, 122, 125, 134, 153, 363).

[Bey20] Tim Beyne. "Block Cipher Invariants as Eigenvectors of Correlation Matrices". In: *Journal of Cryptology* 33.3 (July 2020), pp. 1156–1183 (cit. on p. 115).

[Bey21] Tim Beyne. "A Geometric Approach to Linear Cryptanalysis". In: *ASIACRYPT 2021, Part I*. Ed. by Mehdi Tibouchi and Huaxiong Wang. Vol. 13090. LNCS. Singapore: Springer, Cham, Switzerland, 2021, pp. 36–66 (cit. on pp. 27, 109, 126, 127, 187, 360, 363).

[Bey23] Tim Beyne. "A geometric approach to symmetric-key cryptanalysis". https://cosicdatabase.esat.kuleuven.be/backend/publications/files/these/475. PhD thesis. KU Leuven, 2023 (cit. on p. 51).

[BG05] Olivier Billet and Henri Gilbert. "Resistance of SNOW 2.0 Against Algebraic Attacks". In: *CT-RSA 2005*. Ed. by Alfred Menezes. Vol. 3376. LNCS. San Francisco, CA, USA: Springer, Berlin, Heidelberg, Germany, 2005, pp. 19–28 (cit. on p. 31).

[BGV12] Zvika Brakerski, Craig Gentry, and Vinod Vaikuntanathan. "(Leveled) fully homomorphic encryption without bootstrapping". In: *ITCS 2012*. Ed. by Shafi Goldwasser. Cambridge, MA, USA: ACM, 2012, pp. 309–325 (cit. on p. 281).

[BHK20] Lilya Budaghyan, Tor Helleseth, and Nikolay Kaleyski. "A New Family of APN Quadrinomials". In: *IEEE Transactions on Information Theory* 66.11 (2020), pp. 7081–7087 (cit. on pp. 215, 343, 347).

[BIK23] Lilya Budaghyan, Ivana Ivkovic, and Nikolay S. Kaleyski. "Triplicate functions". In: *Cryptography and Communications* 15.1 (2023), pp. 35–83 (cit. on pp. 190, 191, 195, 217, 220, 223, 230).

[Bir+03] Alex Biryukov, Christophe De Cannière, An Braeken, and Bart Preneel. "A Toolbox for Cryptanalysis: Linear and Affine Equivalence Algorithms". In: *EUROCRYPT 2003*. Ed. by Eli Biham. Vol. 2656. LNCS. Warsaw, Poland: Springer, Berlin, Heidelberg, Germany, 2003, pp. 33–50 (cit. on pp. 166, 167).

[Bir07] Alex Biryukov. "The Design of a Stream Cipher LEX". In: *SAC 2006*. Ed. by Eli Biham and Amr M. Youssef. Vol. 4356. LNCS. Montreal, Canada: Springer, Berlin, Heidelberg, Germany, 2007, pp. 67–75 (cit. on p. 287).

[BL08] Marcus Brinkmann and Gregor Leander. "On the classification of APN functions up to dimension five". In: *Designs, Codes and Cryptography* 49.1 (2008), pp. 273–288. ISSN: 1573-7586 (cit. on pp. 190, 214, 218, 234).

[BL22] Christof Beierle and Gregor Leander. "New Instances of Quadratic APN Functions". In: *IEEE Transactions on Information Theory* 68.1 (2022), pp. 670–678 (cit. on pp. 191, 201, 211, 213, 214, 218, 230–233).

[Bla+99] John Black, Shai Halevi, Hugo Krawczyk, Ted Krovetz, and Phillip Rogaway. "UMAC: Fast and Secure Message Authentication". In: *CRYPTO'99*. Ed. by Michael J. Wiener. Vol. 1666. LNCS. Santa Barbara, CA, USA: Springer, Berlin, Heidelberg, Germany, 1999, pp. 216–233 (cit. on p. 262).

[BN00] Mihir Bellare and Chanathip Namprempre. "Authenticated Encryption: Relations among notions and analysis of the generic composition paradigm". In: *ASIACRYPT 2000*. Ed. by Tatsuaki Okamoto. Vol. 1976. LNCS. Kyoto, Japan: Springer, Berlin, Heidelberg, Germany, 2000, pp. 531–545 (cit. on pp. 7, 76).

[BN13] Céline Blondeau and Kaisa Nyberg. "New Links between Differential and Linear Cryptanalysis". In: *EUROCRYPT 2013*. Ed. by Thomas Johansson and Phong Q. Nguyen. Vol. 7881. LNCS. Athens, Greece: Springer, Berlin, Heidelberg, Germany, 2013, pp. 388–404 (cit. on pp. 52, 360).

[Bon23] Xavier Bonnetain. Personal communication. 2023 (cit. on p. 166).

[Bor+12] Julia Borghoff, Anne Canteaut, Tim Güneysu, Elif Bilge Kavun, Miroslav Knežević, Lars R. Knudsen, Gregor Leander, Ventzislav Nikov, Christof Paar, Christian Rechberger, Peter Rombouts, Søren S. Thomsen, and Tolga Yalçin. "PRINCE - A Low-Latency Block Cipher for Pervasive Computing Applications - Extended Abstract". In: *ASIACRYPT 2012*. Ed. by Xiaoyun Wang and Kazue Sako. Vol. 7658. LNCS. Beijing, China: Springer, Berlin, Heidelberg, Germany, 2012, pp. 208–225 (cit. on pp. 109, 111, 360).

[Bos+17] Alin Bostan, Frédéric Chyzak, Marc Giusti, Romain Lebreton, Grégoire Lecerf, Bruno Salvy, and Éric Schost. *Algorithmes Efficaces en Calcul Formel.* 686 pages. Édition 1.0. Aug. 2017 (cit. on p. 34).

[Bos+22] Jannis Bossert, Eik List, Stefan Lucks, and Sebastian Schmitz. "Pholkos - Efficient Large-State Tweakable Block Ciphers from the AES Round Function". In: *CT-RSA 2022.* Ed. by Steven D. Galbraith. Vol. 13161. LNCS. Virtual Event: Springer, Cham, Switzerland, 2022, pp. 511–536 (cit. on p. 260).

[BOS23] Thibault Balenbois, Jean-Baptiste Orfila, and Nigel P. Smart. "Trivial Transciphering With Trivium and TFHE". In: *Proceedings of the 11th Workshop on Encrypted Computing & Applied Homomorphic Cryptography, Copenhagen, Denmark, 26 November 2023.* Ed. by Michael Brenner, Anamaria Costache, and Kurt Rohloff. ACM, 2023, pp. 69–78 (cit. on pp. 282, 299).

[Bou+10] Charles Bouillaguet, Orr Dunkelman, Gaëtan Leurent, and Pierre-Alain Fouque. "Another Look at Complementation Properties". In: *FSE 2010.* Ed. by Seokhie Hong and Tetsu Iwata. Vol. 6147. LNCS. Seoul, Korea: Springer, Berlin, Heidelberg, Germany, 2010, pp. 347–364 (cit. on pp. 155, 160, 161, 364).

[Bou23] Clémence Bouvier. "Cryptanalysis and design of symmetric primitives defined over large finite fields". `https://inria.hal.science/tel-04327955v2/file/BOUVIER_Clemence_these_2023.pdf`. PhD thesis. Sorbonne Université, Nov. 2023 (cit. on p. 35).

[BP17] Alex Biryukov and Leo Perrin. *State of the Art in Lightweight Symmetric Cryptography.* Cryptology ePrint Archive, Report 2017/511. 2017 (cit. on pp. 8, 109, 360).

[BPR24] Nicolas Bon, David Pointcheval, and Matthieu Rivain. "Optimized Homomorphic Evaluation of Boolean Functions". In: *IACR Transactions on Cryptographic Hardware and Embedded Systems* 2024.3 (2024), pp. 302–341 (cit. on p. 283).

[BPU16] Alex Biryukov, Léo Perrin, and Aleksei Udovenko. "Reverse-Engineering the S-Box of Streebog, Kuznyechik and STRIBOBr1". In: *EUROCRYPT 2016, Part I.* Ed. by Marc Fischlin and Jean-Sébastien Coron. Vol. 9665. LNCS. Vienna, Austria: Springer, Berlin, Heidelberg, Germany, 2016, pp. 372–402 (cit. on pp. 255, 353, 366).

[BPW23] Alexander Bors, Daniel Panario, and Qiang Wang. *Functional graphs of generalized cyclotomic mappings of finite fields.* arXiv 1108.1873. 2023 (cit. on p. 198).

[BR22] Tim Beyne and Vincent Rijmen. “Differential Cryptanalysis in the Fixed-Key Model”. In: *CRYPTO 2022, Part III.* Ed. by Yevgeniy Dodis and Thomas Shrimpton. Vol. 13509. LNCS. Santa Barbara, CA, USA: Springer, Cham, Switzerland, 2022, pp. 687–716 (cit. on pp. 44, 360).

[Bra+08] Carl Bracken, Eimear Byrne, Nadya Markin, and Gary McGuire. “New families of quadratic almost perfect nonlinear trinomials and multinomials”. In: *Finite Fields and Their Applications* 14.3 (2008), pp. 703–714 (cit. on pp. 215, 217).

[Bra+11a] Carl Bracken, Eimear Byrne, Nadya Markin, and Gary McGuire. “A few more quadratic APN functions”. In: *Cryptography and Communications* 3.1 (2011), pp. 43–53 (cit. on pp. 215, 343, 346).

[Bra+11b] Carl Bracken, Eimear Byrne, Gary McGuire, and Gabriele Nebe. “On the equivalence of quadratic APN functions”. In: *Designs, Codes and Cryptography* 61.3 (2011), pp. 261–272 (cit. on p. 218).

[Bro+10] K. A. Browning, J.F. Dillon, M. T. McQuistan, and A. J. Wolfe. “An APN Permutation in Dimension Six”. In: *Post-proceedings of the 9-th International Conference on Finite Fields Their Appl.* Vol. 518. American Mathematical Society, 2010, pp. 33–42 (cit. on pp. 189, 190, 202, 248, 249, 365).

[Bry89] Lennart Brynielsson. “A short proof of the Xiao-Massey lemma”. In: *IEEE Transactions on Information Theory* 35.6 (1989), p. 1344 (cit. on p. 293).

[BS01] Alex Biryukov and Adi Shamir. “Structural Cryptanalysis of SASAS”. In: *EUROCRYPT 2001.* Ed. by Birgit Pfitzmann. Vol. 2045. LNCS. Innsbruck, Austria: Springer, Berlin, Heidelberg, Germany, 2001, pp. 394–405 (cit. on p. 70).

[BS91a] Eli Biham and Adi Shamir. “Differential Cryptanalysis of DES-like Cryptosystems”. In: *Journal of Cryptology* 4.1 (Jan. 1991), pp. 3–72 (cit. on pp. 15, 36, 360).

[BS91b] Eli Biham and Adi Shamir. “Differential Cryptanalysis of DES-like Cryptosystems”. In: *CRYPTO’90.* Ed. by Alfred J. Menezes and Scott A. Vanstone. Vol. 537. LNCS. Santa Barbara, CA, USA: Springer, Berlin, Heidelberg, Germany, 1991, pp. 2–21 (cit. on pp. 15, 36, 45, 360).

[BS93] Eli Biham and Adi Shamir. “Differential Cryptanalysis of the Full 16-Round DES”. In: *CRYPTO’92.* Ed. by Ernest F. Brickell. Vol. 740. LNCS. Santa Barbara, CA, USA: Springer, Berlin, Heidelberg, Germany, 1993, pp. 487–496 (cit. on p. 36).

[BSV07] Thomas Baignères, Jacques Stern, and Serge Vaudenay. “Linear Cryptanalysis of Non Binary Ciphers”. In: *SAC 2007*. Ed. by Carlisle M. Adams, Ali Miri, and Michael J. Wiener. Vol. 4876. LNCS. Ottawa, Canada: Springer, Berlin, Heidelberg, Germany, 2007, pp. 184–211 (cit. on p. 295).

[Bud+06] Lilya Budaghyan, Claude Carlet, Patrick Felke, and Gregor Leander. “An infinite class of quadratic APN functions which are not equivalent to power mappings”. In: *2006 IEEE International Symposium on Information Theory*. 2006, pp. 2637–2641 (cit. on pp. 215, 346).

[Bud+17] Lilya Budaghyan, Tor Helleseth, Nian Li, and Bo Sun. “Some Results on the Known Classes of Quadratic APN Functions”. In: *Codes, Cryptology and Information Security - C2SI 2017*. Ed. by Said El Hajji, Abderrahmane Nitaj, and El Mamoun Souidi. Vol. 10194. Lecture Notes in Computer Science. Springer, 2017, pp. 3–16 (cit. on p. 246).

[Bud+20] Lilya Budaghyan, Marco Calderini, Claude Carlet, Robert S. Coulter, and Irene Villa. “Constructing APN Functions Through Isotopic Shifts”. In: *IEEE Transactions on Information Theory* 66.8 (2020), pp. 5299–5309 (cit. on pp. 215, 343, 347).

[Bud+22] Lilya Budaghyan, Marco Calderini, Claude Carlet, Diana Davidova, and Nikolay S. Kaleyski. “On Two Fundamental Problems on APN Power Functions”. In: *IEEE Transactions on Information Theory* 68.5 (2022), pp. 3389–3403 (cit. on p. 241).

[BV23] Tim Beyne and Michiel Verbauwhede. “Integral Cryptanalysis Using Algebraic Transition Matrices”. In: *IACR Transactions on Symmetric Cryptology* 2023.4 (2023), pp. 244–269 (cit. on p. 72).

[BW22] Alexander Bors and Qiang Wang. “Generalized Cyclotomic Mappings: Switching Between Polynomial, Cyclotomic, and Wreath Product Form”. In: *Communications in Mathematical Research* 38.2 (2022), pp. 246–318 (cit. on p. 202).

[Cae13] *CAESAR: Competition for Authenticated Encryption: Security, Applicability, and Robustness*. `https://competitions.cr.yp.to/caesar.html`. Jan. 2013 (cit. on pp. 2, 4, 63, 75, 173, 360, 366).

[Can+15] Anne Canteaut, Thomas Fuhr, Henri Gilbert, María Naya-Plasencia, and Jean-René Reinhard. “Multiple Differential Cryptanalysis of Round-Reduced PRINCE”. In: *FSE 2014*. Ed. by Carlos Cid and Christian Rechberger. Vol. 8540. LNCS. London, UK: Springer, Berlin, Heidelberg, Germany, 2015, pp. 591–610 (cit. on p. 176).

[Can16] Anne Canteaut. *Lecture Notes on Cryptographic Boolean Functions*. `https://www.rocq.inria.fr/secret/Anne.Canteaut/poly.pdf`. 2016 (cit. on pp. 19, 24, 48, 53, 361).

[Can+16] Anne Canteaut, Sergiu Carpov, Caroline Fontaine, Tancrède Lepoint, María Naya-Plasencia, Pascal Paillier, and Renaud Sirdey. "Stream Ciphers: A Practical Solution for Efficient Homomorphic-Ciphertext Compression". In: *FSE 2016*. Ed. by Thomas Peyrin. Vol. 9783. LNCS. Bochum, Germany: Springer, Berlin, Heidelberg, Germany, 2016, pp. 313–333 (cit. on pp. 258, 282).

[Can+17] Anne Canteaut, Eran Lambooij, Samuel Neves, Shahram Rasoolzadeh, Yu Sasaki, and Marc Stevens. "Refined Probability of Differential Characteristics Including Dependency Between Multiple Rounds". In: *IACR Transactions on Symmetric Cryptology* 2017.2 (2017), pp. 203–227. ISSN: 2519-173X (cit. on p. 44).

[Can+23] Federico Canale, Tim Güneysu, Gregor Leander, Jan Philipp Thoma, Yosuke Todo, and Rei Ueno. "SCARF - A Low-Latency Block Cipher for Secure Cache-Randomization". In: *32nd USENIX Security Symposium, USENIX Security 2023, Anaheim, CA, USA, August 9-11, 2023*. Ed. by Joseph A. Calandrino and Carmela Troncoso. USENIX Association, 2023, pp. 1937–1954 (cit. on p. 7).

[Car11] Claude Carlet. "Relating three nonlinearity parameters of vectorial functions and building APN functions from bent functions". In: *Designs, Codes and Cryptography* 59.1-3 (2011), pp. 89–109 (cit. on p. 217).

[Car15] Claude Carlet. "Open Questions on Nonlinearity and on APN Functions". In: *Arithmetic of Finite Fields*. Ed. by Çetin Kaya Koç, Sihem Mesnager, and Erkay Savaş. Cham: Springer International Publishing, 2015, pp. 83–107. ISBN: 978-3-319-16277-5 (cit. on pp. 198, 246, 360, 365).

[Car21] Claude Carlet. *Boolean Functions for Cryptography and Coding Theory*. Cambridge University Press, 2021 (cit. on pp. 19, 30, 54, 55, 238, 357, 361).

[CBC21] Marco Calderini, Lilya Budaghyan, and Claude Carlet. "On known constructions of APN and AB functions and their relation to each other". In: *Rad Hrvatske akademije znanosti i umjetnosti. Matematičke znanosti* 546= 25 (2021), pp. 79–105 (cit. on pp. 216, 217, 344, 348).

[CBS19] Roberto Civino, Céline Blondeau, and Massimiliano Sala. "Differential attacks: using alternative operations". In: *Designs, Codes and Cryptography* 87.2-3 (2019), pp. 225–247 (cit. on pp. 110, 135, 138, 141–143, 147, 151, 154, 364).

[CC23] Li-An Chen and Robert S. Coulter. "Bounds on the differential uniformity of the Wan-Lidl polynomials". In: *Cryptography and Communications* 15.6 (2023), pp. 1069–1085 (cit. on pp. 198, 241).

[CCD00] Anne Canteaut, Pascale Charpin, and Hans Dobbertin. “Weight Divisibility of Cyclic Codes, Highly Nonlinear Functions on $\mathbb{F}_{2^m}$, and Crosscorrelation of Maximum-Length Sequences”. In: *SIAM J. Discret. Math.* 13.1 (2000), pp. 105–138 (cit. on pp. 236, 241).

[CCI24a] Marco Calderini, Roberto Civino, and Riccardo Invernizzi. *Optimal s-boxes against alternative operations.* 2024. arXiv: 2403.20059 [cs.CR] (cit. on pp. 135, 138, 141–143, 147, 151, 152).

[CCI24b] Marco Calderini, Roberto Civino, and Riccardo Invernizzi. “Optimal s-boxes against alternative operations”. In: *WCC 2024: The Thirteenth International Workshop on Coding and Cryptography.* https://wcc2024.sites.dmi.unipg.it/. Perugia, Italy, 2024, pp. 87–98 (cit. on pp. 135, 138, 141–143, 147, 151, 152).

[CCP22] Anne Canteaut, Alain Couvreur, and Léo Perrin. “Recovering or testing extended-affine equivalence”. In: *IEEE Transactions on Information Theory* 68.9 (2022), pp. 6187–6206 (cit. on pp. 37, 228–230).

[CCP24] Anne Canteaut, Alain Couvreur, and Léo Perrin. “On the Properties of the Ortho-Derivatives of Quadratic Functions”. In: *WCC 2024: The Thirteenth International Workshop on Coding and Cryptography.* https://wcc2024.sites.dmi.unipg.it/WCC_proceedings.pdf. Perugia, Italy, 2024, pp. 99–110 (cit. on p. 228).

[CCS21] Marco Calderini, Roberto Civino, and Massimiliano Sala. “On properties of translation groups in the affine general linear group with applications to cryptography”. In: *Journal of Algebra* 569 (2021), pp. 658–680. ISSN: 0021-8693 (cit. on pp. 138, 141, 143, 146).

[CCZ98] Claude Carlet, Pascale Charpin, and Victor Zinoviev. “Codes, bent functions and permutations suitable for DES-like cryptosystems”. In: *Designs, Codes and Cryptography* 15 (1998), pp. 125–156 (cit. on pp. 57, 236).

[CDP17] Anne Canteaut, Sébastien Duval, and Léo Perrin. “A Generalisation of Dillon’s APN Permutation With the Best Known Differential and Nonlinear Properties for All Fields of Size 2^{4k+2}”. In: *IEEE Transactions on Information Theory* 63.11 (2017), pp. 7575–7591. ISSN: 0018-9448 (cit. on p. 200).

[CDVS05] Andrea Caranti, Francesca Dalla Volta, and Massimiliano Sala. “Abelian regular subgroups of the affine group and radical rings”. In: *Publicationes Mathematicae* 69 (Nov. 2005) (cit. on pp. 138, 141, 147).

[CF08] Claude Carlet and Keqin Feng. “An Infinite Class of Balanced Functions with Optimal Algebraic Immunity, Good Immunity to Fast Algebraic Attacks and Good Nonlinearity”. In: *ASIACRYPT 2008*. Ed. by Josef Pieprzyk. Vol. 5350. LNCS. Melbourne, Australia: Springer, Berlin, Heidelberg, Germany, 2008, pp. 425–440 (cit. on p. 357).

[CGT16] Claude Carlet, Guang Gong, and Yin Tan. “Quadratic zero-difference balanced functions, APN functions and strongly regular graphs”. In: *Designs, Codes and Cryptography* 78.3 (2016), pp. 629–654 (cit. on p. 241).

[Cha+17] Colin Chaigneau, Thomas Fuhr, Henri Gilbert, Jérémy Jean, and Jean-René Reinhard. “Cryptanalysis of NORX v2.0”. In: *IACR Transactions on Symmetric Cryptology* 2017.1 (2017), pp. 156–174. ISSN: 2519-173X (cit. on pp. 109, 155, 160, 162).

[Chi+16] Ilaria Chillotti, Nicolas Gama, Mariya Georgieva, and Malika Izabachène. “Faster Fully Homomorphic Encryption: Bootstrapping in Less Than 0.1 Seconds”. In: *ASIACRYPT 2016, Part I*. Ed. by Jung Hee Cheon and Tsuyoshi Takagi. Vol. 10031. LNCS. Hanoi, Vietnam: Springer, Berlin, Heidelberg, Germany, 2016, pp. 3–33 (cit. on pp. 258, 280, 281).

[Chi+17] Ilaria Chillotti, Nicolas Gama, Mariya Georgieva, and Malika Izabachène. “Faster Packed Homomorphic Operations and Efficient Circuit Bootstrapping for TFHE”. In: *ASIACRYPT 2017, Part I*. Ed. by Tsuyoshi Takagi and Thomas Peyrin. Vol. 10624. LNCS. Hong Kong, China: Springer, Cham, Switzerland, 2017, pp. 377–408 (cit. on pp. 258, 280, 281).

[Chi+20] Ilaria Chillotti, Nicolas Gama, Mariya Georgieva, and Malika Izabachène. “TFHE: Fast Fully Homomorphic Encryption Over the Torus”. In: *Journal of Cryptology* 33.1 (Jan. 2020), pp. 34–91 (cit. on pp. 258, 280, 281, 367).

[CHK22] Donghoon Chang, Deukjo Hong, and Jinkeon Kang. *Conditional Cube Attacks on Ascon-128 and Ascon-80pq in a Nonce-misuse Setting*. Cryptology ePrint Archive, Report 2022/544. 2022 (cit. on pp. 80, 84, 85, 101, 104, 105).

[Cho+24] Mingyu Cho, Woohyuk Chung, Jincheol Ha, Jooyoung Lee, Eun-Gyeol Oh, and Mincheol Son. “FRAST: TFHE-Friendly Cipher Based on Random S-Boxes”. In: *IACR Transactions on Symmetric Cryptology* 2024.3 (2024), pp. 1–43 (cit. on pp. 282, 299).

[CHP17] Claude Carlet, Annelie Heuser, and Stjepan Picek. “Trade-Offs for S-Boxes: Cryptographic Properties and Side-Channel Resilience”. In: *Applied Cryptography and Network Security - ACNS 2017*. Ed. by Dieter Gollmann, Atsuko Miyaji, and Hiroaki Kikuchi. Vol. 10355.

Lecture Notes in Computer Science. Springer, 2017, pp. 393–414 (cit. on pp. 238, 239).

[Cid+22] Carlos Cid, Lorenzo Grassi, Aldo Gunsing, Reinhard Lüftenegger, Christian Rechberger, and Markus Schofnegger. “Influence of the Linear Layer on the Algebraic Degree in SP-Networks”. In: *IACR Transactions on Symmetric Cryptology* 2022.1 (2022), pp. 110–137 (cit. on p. 32).

[CJS01] Vladimor V. Chepyzhov, Thomas Johansson, and Ben J. M. Smeets. “A Simple Algorithm for Fast Correlation Attacks on Stream Ciphers”. In: *FSE 2000*. Ed. by Bruce Schneier. Vol. 1978. LNCS. New York, NY, USA: Springer, Berlin, Heidelberg, Germany, 2001, pp. 181–195 (cit. on p. 295).

[CK91] David G. Cantor and Erich L. Kaltofen. “On Fast Multiplication of Polynomials over Arbitrary Algebras”. In: *Acta Informatica* 28.7 (1991), pp. 693–701 (cit. on p. 34).

[CL21] Benjamin Chase and Petr Lisonek. “Kim-type APN functions are affine equivalent to Gold functions”. In: *Cryptography and Communications* 13.6 (2021), pp. 981–993 (cit. on pp. 198, 246).

[CLV22] Marco Calderini, Kangquan Li, and Irene Villa. “Two new families of bivariate APN functions”. In: *arXiv preprint arXiv:2204.07462* (2022) (cit. on pp. 216, 218, 344, 349).

[CM10] Claude Carlet and Sihem Mesnager. *A note on semi-bent Boolean functions*. Cryptology ePrint Archive, Paper 2010/486. `https://eprint.iacr.org/2010/486`. 2010 (cit. on p. 357).

[Cos+22] Orel Cosseron, Clément Hoffmann, Pierrick Méaux, and François-Xavier Standaert. “Towards Case-Optimized Hybrid Homomorphic Encryption - Featuring the Elisabeth Stream Cipher”. In: *ASIACRYPT 2022, Part III*. Ed. by Shweta Agrawal and Dongdai Lin. Vol. 13793. LNCS. Taipei, Taiwan: Springer, Cham, Switzerland, 2022, pp. 32–67 (cit. on p. 282).

[CP19] Anne Canteaut and Léo Perrin. “On CCZ-equivalence, extended-affine equivalence, and function twisting”. In: *Finite Fields and Their Applications* 56 (2019), pp. 209–246. ISSN: 1071-5797 (cit. on pp. 58, 61, 190, 201, 248).

[CPT19] Anne Canteaut, Léo Perrin, and Shizhu Tian. “If a generalised butterfly is APN then it operates on 6 bits”. In: *Cryptography and Communications* 11.6 (2019), pp. 1147–1164 (cit. on p. 200).

[CS16] Benoît Cogliati and Yannick Seurin. “EWCDM: An Efficient, Beyond-Birthday Secure, Nonce-Misuse Resistant MAC”. In: *CRYPTO 2016, Part I*. Ed. by Matthew Robshaw and Jonathan Katz. Vol. 9814. LNCS. Santa Barbara, CA, USA: Springer, Berlin, Heidelberg, Germany, 2016, pp. 121–149 (cit. on pp. 258, 262, 279, 366).

[CT00] Anne Canteaut and Michaël Trabbia. “Improved Fast Correlation Attacks Using Parity-Check Equations of Weight 4 and 5”. In: *EUROCRYPT 2000*. Ed. by Bart Preneel. Vol. 1807. LNCS. Bruges, Belgium: Springer, Berlin, Heidelberg, Germany, 2000, pp. 573–588 (cit. on pp. 14, 295).

[CU57] Leonard Carlitz and Saburo Uchiyama. “Bounds for exponential sums”. In: *Duke Mathematical Journal* 24.1 (1957), pp. 37 –41 (cit. on p. 189).

[CV02] Anne Canteaut and Marion Videau. “Degree of Composition of Highly Nonlinear Functions and Applications to Higher Order Differential Cryptanalysis”. In: *EUROCRYPT 2002*. Ed. by Lars R. Knudsen. Vol. 2332. LNCS. Amsterdam, The Netherlands: Springer, Berlin, Heidelberg, Germany, 2002, pp. 518–533 (cit. on p. 32).

[CV95] Florent Chabaud and Serge Vaudenay. “Links Between Differential and Linear Cryptanalysis”. In: *EUROCRYPT’94*. Ed. by Alfredo De Santis. Vol. 950. LNCS. Perugia, Italy: Springer, Berlin, Heidelberg, Germany, 1995, pp. 356–365 (cit. on pp. 52, 56, 236, 360).

[CW77] J Lawrence Carter and Mark N Wegman. “Universal classes of hash functions”. In: *Proceedings of the ninth annual ACM symposium on Theory of computing*. ACM. 1977, pp. 106–112 (cit. on pp. 262, 279).

[Cze20] Ingo Czerwinski. *On the minimal value set size of APN functions*. Cryptology ePrint Archive, Report 2020/705. 2020 (cit. on pp. 238, 239).

[DC98] Joan Daemen and Craig S. K. Clapp. “Fast Hashing and Stream Encryption with PANAMA”. In: *FSE’98*. Ed. by Serge Vaudenay. Vol. 1372. LNCS. Paris, France: Springer, Berlin, Heidelberg, Germany, 1998, pp. 60–74 (cit. on p. 265).

[DDS14] Itai Dinur, Orr Dunkelman, and Adi Shamir. “Improved Practical Attacks on Round-Reduced Keccak”. In: *Journal of Cryptology* 27.2 (Apr. 2014), pp. 183–209 (cit. on p. 44).

[Des] *Data Encryption Standard*. National Bureau of Standards, NBS FIPS PUB 46, U.S. Department of Commerce. Jan. 1977 (cit. on pp. 2, 3).

[DF04] David Steven Dummit and Richard M. Foote. *Abstract Algebra*. Third. Hoboken, NJ: John Wiley & sons, 2004. ISBN: 0-471-43334-9 (cit. on p. 270).

[DGV95] Joan Daemen, René Govaerts, and Joos Vandewalle. “Correlation Matrices”. In: *FSE’94*. Ed. by Bart Preneel. Vol. 1008. LNCS. Leuven, Belgium: Springer, Berlin, Heidelberg, Germany, 1995, pp. 275–285 (cit. on pp. 27, 50).

[DH76] Whitfield Diffie and Martin E. Hellman. “New Directions in Cryptography”. In: *IEEE Transactions on Information Theory* 22.6 (1976), pp. 644–654 (cit. on pp. 2, 3).

[Dil09] John Dillon. “APN polynomials: an update”. In: *International Conference on Finite fields and applications - Fq9*. 2009 (cit. on pp. 189, 231).

[Din18] Itai Dinur. “An Improved Affine Equivalence Algorithm for Random Permutations”. In: *EUROCRYPT 2018, Part I*. Ed. by Jesper Buus Nielsen and Vincent Rijmen. Vol. 10820. LNCS. Tel Aviv, Israel: Springer, Cham, Switzerland, 2018, pp. 413–442 (cit. on p. 166).

[Dix71] John D Dixon. “Maximal Abelian subgroups of the symmetric groups”. In: *Canadian Journal of Mathematics* 23.3 (1971), pp. 426–438 (cit. on p. 139).

[DKR97] Joan Daemen, Lars R. Knudsen, and Vincent Rijmen. “The Block Cipher Square”. In: *FSE’97*. Ed. by Eli Biham. Vol. 1267. LNCS. Haifa, Israel: Springer, Berlin, Heidelberg, Germany, 1997, pp. 149–165 (cit. on p. 69).

[DMN23] Christoph Dobraunig, Bart Mennink, and Samuel Neves. “EliMAC: Speeding Up LightMAC by around 20%”. In: *IACR Transactions on Symmetric Cryptology* 2023.2 (2023), pp. 69–93 (cit. on p. 262).

[Dob01] Hans Dobbertin. “Almost Perfect Nonlinear Power Functions on GF(2^n): A New Case for n Divisible by 5”. In: *Finite Fields and Applications*. Ed. by Dieter Jungnickel and Harald Niederreiter. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 113–121. ISBN: 978-3-642-56755-1 (cit. on pp. 189, 190, 241, 365).

[Dob+15] Christoph Dobraunig, Maria Eichlseder, Florian Mendel, and Martin Schläffer. “Cryptanalysis of Ascon”. In: *CT-RSA 2015*. Ed. by Kaisa Nyberg. Vol. 9048. LNCS. San Francisco, CA, USA: Springer, Cham, Switzerland, 2015, pp. 371–387 (cit. on pp. 79–81, 103).

[Dob+16] Christoph Dobraunig, Maria Eichlseder, Florian Mendel, and Martin Schläffer. *Ascon v1.2*. Tech. rep. `http://competitions.cr.yp.to/round3/asconv12.pdf`. CAESAR: Competition for Authenticated Encryption: Security, Applicability, and Robustness, 2016 (cit. on pp. 2, 4).

[Dob+19] Christoph Dobraunig, Maria Eichlseder, Florian Mendel, and Martin Schläffer. *Ascon v1.2*. Tech. rep. `https://csrc.nist.gov/Projects/lightweight-cryptography/finalists`. National Institute of Standards and Technology, 2019 (cit. on pp. 2, 4, 359).

[Dob+21] Christoph Dobraunig, Maria Eichlseder, Florian Mendel, and Martin Schläffer. “Ascon v1.2: Lightweight Authenticated Encryption and Hashing”. In: *Journal of Cryptology* 34.3 (July 2021), p. 33 (cit. on pp. 4, 33, 61–63, 75, 359, 360).

[Dob98] Hans Dobbertin. “Cryptanalysis of MD4”. In: *Journal of Cryptology* 11.4 (Sept. 1998), pp. 253–271 (cit. on p. 44).

[Dob99a] Hans Dobbertin. “Almost Perfect Nonlinear Power Functions on GF(2^n): The Niho Case”. In: *Inf. Comput.* 151.1-2 (1999), pp. 57–72 (cit. on pp. 189, 365).

[Dob99b] Hans Dobbertin. “Almost Perfect Nonlinear Power Functions on GF(2^n): The Welch Case”. In: *IEEE Transactions on Information Theory* 45.4 (1999), pp. 1271–1275 (cit. on pp. 189, 365).

[DR01] Joan Daemen and Vincent Rijmen. “The Wide Trail Design Strategy”. In: *8th IMA International Conference on Cryptography and Coding*. Ed. by Bahram Honary. Vol. 2260. LNCS. Cirencester, UK: Springer, Berlin, Heidelberg, Germany, 2001, pp. 222–238 (cit. on p. 45).

[DR02] Joan Daemen and Vincent Rijmen. *The Design of Rijndael: AES - The Advanced Encryption Standard*. Information Security and Cryptography. Springer, 2002. ISBN: 3-540-42580-2 (cit. on pp. 2, 4, 33, 45, 260, 359).

[DS09] Itai Dinur and Adi Shamir. “Cube Attacks on Tweakable Black Box Polynomials”. In: *EUROCRYPT 2009*. Ed. by Antoine Joux. Vol. 5479. LNCS. Cologne, Germany: Springer, Berlin, Heidelberg, Germany, 2009, pp. 278–299 (cit. on pp. 14, 63, 66, 72, 74, 361).

[Dun+24] Orr Dunkelman, Shibam Ghosh, Nathan Keller, Gaëtan Leurent, Avichai Marmor, and Victor Mollimard. “Partial Sums Meet FFT: Improved Attack on 6-Round AES”. In: *EUROCRYPT 2024, Part I*. Ed. by Marc Joye and Gregor Leander. Vol. 14651. LNCS. Zurich, Switzerland: Springer, Cham, Switzerland, 2024, pp. 128–157 (cit. on p. 70).

[Dwo04] Morris Dworkin. *Recommendation for Block Cipher Modes of Operation: The CCM Mode for Authentication and Confidentiality*. Tech. rep. NIST Special Publication 800-38C. National Institute of Standards and Technology, May 2004 (cit. on p. 13).

[Dwo07] Morris Dworkin. *Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC*. Tech. rep. NIST Special Publication 800-38D. National Institute of Standards and Technology, Nov. 2007 (cit. on pp. 13, 262).

[Ecr04] *eSTREAM: the ECRYPT Stream Cipher Project*. `https://www.ecrypt.eu.org/stream/`. Nov. 2004 (cit. on p. 4).

[EKP06] Yves Edel, Gohar M. Kyureghyan, and Alexander Pott. “A new APN function which is not equivalent to a power mapping”. In: *IEEE Transactions on Information Theory* 52.2 (2006), pp. 744–747 (cit. on pp. 189, 190).

[Ell+20] Pål Ellingsen, Patrick Felke, Constanza Riera, Pantelimon Stănică, and Anton Tkachenko. “C-Differentials, Multiplicative Uniformity, and (Almost) Perfect c-Nonlinearity”. In: *IEEE Transactions on Information Theory* 66.9 (2020), pp. 5781–5789 (cit. on p. 160).

[Ell+21] Pål Ellingsen, Constanza Riera, Pantelimon Stănică, and Anton Tkachenko. “An Extension of the Avalanche Criterion in the Context of c-Differentials”. In: *Proceedings of the 18th International Conference on Security and Cryptography - Volume 1: SECRYPT,* INSTICC. SciTePress, 2021, pp. 460–467. ISBN: 978-989-758-524-1 (cit. on p. 160).

[EP09] Yves Edel and Alexander Pott. “A new almost perfect nonlinear function which is not quadratic”. In: *Adv. Math. Commun.* 3.1 (2009), pp. 59–81 (cit. on pp. 190, 218, 234).

[EU06] Larry Ewing and User:Lunkwill. *Tux ECB Wikipedia page.* https://commons.wikimedia.org/wiki/File:Tux_ecb.jpg. 2006 (cit. on p. 13).

[Fed12] Federal Agency on Technical Regulation and Metrology. *Information Technology – Data Security: Hash function.* English version available at http://wwwold.tc26.ru/en/standard/gost/GOST_R_34_11-2012_eng.pdf. 2012 (cit. on pp. 255, 351, 353, 366).

[Fed15] Federal Agency on Technical Regulation and Metrology. *Information Technology – Data Security: Block ciphers.* English version available at http://wwwold.tc26.ru/en/standard/gost/GOST_R_34_12_2015_ENG.pdf. 2015 (cit. on pp. 255, 351, 353, 366).

[Fer+01] Niels Ferguson, John Kelsey, Stefan Lucks, Bruce Schneier, Michael Stay, David Wagner, and Doug Whiting. “Improved Cryptanalysis of Rijndael”. In: *FSE 2000.* Ed. by Bruce Schneier. Vol. 1978. LNCS. New York, NY, USA: Springer, Berlin, Heidelberg, Germany, 2001, pp. 213–230 (cit. on p. 70).

[FFW17] Shihui Fu, Xiutao Feng, and Baofeng Wu. “Differentially 4-Uniform Permutations with the Best Known Nonlinearity from Butterflies”. In: *IACR Transactions on Symmetric Cryptology* 2017.2 (2017), pp. 228–249. ISSN: 2519-173X (cit. on p. 200).

[FG22] Antonio Florez Gutierrez. “Improved Techniques in the Cryptanalysis of Symmetric Primitives”. https://inria.hal.science/tel-03878739v2/file/FLOREZ_GUTIERREZ_Antonio_2022.pdf. PhD thesis. Sorbonne Université, Sept. 2022 (cit. on p. 51).

[Fog22] Agner Fog. *Lists of instruction latencies, throughputs and micro-operation breakdowns for Intel, AMD, and VIA CPUs.* 2022 (cit. on p. 264).

[Gan90] Felix Ruvimovich Gantmacher. *The Theory of Matrices.* Second. Vol. 1. Chelsea Publishing Company, 1990 (cit. on p. 270).

[Gen09] Craig Gentry. “A fully homomorphic encryption scheme”. crypto.stanford.edu/craig. PhD thesis. Stanford University, 2009 (cit. on pp. 282, 367).

[GHS12] Craig Gentry, Shai Halevi, and Nigel P. Smart. “Homomorphic Evaluation of the AES Circuit”. In: *CRYPTO 2012*. Ed. by Reihaneh Safavi-Naini and Ran Canetti. Vol. 7417. LNCS. Santa Barbara, CA, USA: Springer, Berlin, Heidelberg, Germany, 2012, pp. 850–867 (cit. on p. 282).

[Gil+23] Henri Gilbert, Rachelle Heim Boissier, Jérémy Jean, and Jean-René Reinhard. “Cryptanalysis of Elisabeth-4”. In: *ASIACRYPT 2023, Part III*. Ed. by Jian Guo and Ron Steinfeld. Vol. 14440. LNCS. Guangzhou, China: Springer, Singapore, Singapore, 2023, pp. 256–284 (cit. on pp. 31, 282).

[GK08] Shay Gueron and Michael E. Kounavis. “Vortex: A New Family of One-Way Hash Functions Based on AES Rounds and Carry-Less Multiplication”. In: *Information Security*. Ed. by Tzong-Chen Wu, Chin-Laung Lei, Vincent Rijmen, and Der-Tsai Lee. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 331–340. ISBN: 978-3-540-85886-7 (cit. on p. 260).

[GK21] Faruk Göloğlu and Lukas Kölsch. “Equivalences of biprojective almost perfect nonlinear functions”. In: *arXiv preprint arXiv:2111.04197* (2021) (cit. on pp. 216, 217, 344, 348).

[GL16] David Gérault and Pascal Lafourcade. “Related-Key Cryptanalysis of Midori”. In: *INDOCRYPT 2016*. Ed. by Orr Dunkelman and Somitra Kumar Sanadhya. Vol. 10095. LNCS. Kolkata, India: Springer, Cham, Switzerland, 2016, pp. 287–304 (cit. on p. 109).

[GL20] Faruk Göloğlu and Philippe Langevin. “Almost perfect nonlinear families which are not equivalent to permutations”. In: *Finite Fields and Their Applications* 67 (2020), p. 101707. ISSN: 1071-5797 (cit. on p. 246).

[GM16] Shay Gueron and Nicky Mouha. “Simpira v2: A Family of Efficient Permutations Using the AES Round Function”. In: *ASIACRYPT 2016, Part I*. Ed. by Jung Hee Cheon and Tsuyoshi Takagi. Vol. 10031. LNCS. Hanoi, Vietnam: Springer, Berlin, Heidelberg, Germany, 2016, pp. 95–125 (cit. on p. 260).

[Göl15] Faruk Göloğlu. “Almost perfect nonlinear trinomials and hexanomials”. In: *Finite Fields and Their Applications* 33 (2015), pp. 258–282. ISSN: 1071-5797 (cit. on pp. 197, 203, 245, 246, 357).

[Göl22] Faruk Göloğlu. “Biprojective Almost Perfect Nonlinear Functions”. In: *IEEE Transactions on Information Theory* 68.7 (2022), pp. 4750–4760 (cit. on pp. 200, 209, 216, 217, 219, 344, 348).

[Göl23] Faruk Göloğlu. “Classification of (q, q)-Biprojective APN Functions”. In: *IEEE Transactions on Information Theory* 69.3 (2023), pp. 1988–1999 (cit. on pp. 198, 200, 209, 245, 246, 253).

[Gol68] Robert Gold. “Maximal recursive sequences with 3-valued recursive cross-correlation functions (Corresp.)” In: *IEEE Transactions on Information Theory* 14.1 (1968), pp. 154–156 (cit. on p. 189).

[Gol97] Jovan Dj. Golic. “Cryptanalysis of Alleged A5 Stream Cipher”. In: *EUROCRYPT’97*. Ed. by Walter Fumy. Vol. 1233. LNCS. Konstanz, Germany: Springer, Berlin, Heidelberg, Germany, 1997, pp. 239–255 (cit. on p. 287).

[Gro+15a] Vincent Grosso, Gaëtan Leurent, François-Xavier Standaert, and Kerem Varici. “LS-Designs: Bitslice Encryption for Efficient Masked Software Implementations”. In: *FSE 2014*. Ed. by Carlos Cid and Christian Rechberger. Vol. 8540. LNCS. London, UK: Springer, Berlin, Heidelberg, Germany, 2015, pp. 18–37 (cit. on p. 169).

[Gro+15b] Vincent Grosso, Gaëtan Leurent, François-Xavier Standaert, Kerem Varici, Anthony Journault, François Durvaux, Lubos Gaspar, and Stéphanie Kerckhof. *SCREAM v3. Submission to CAESAR Competition*. 2015 (cit. on p. 173).

[Gue08] Shay Gueron. *Intel Advanced Encryption Standard (AES) New Instructions Set*. White Paper Rev 3.01 (09/2012). Intel Corporation, 2008 (cit. on pp. 257, 260, 366).

[Guo+16] Jian Guo, Jérémy Jean, Ivica Nikolic, Kexin Qiao, Yu Sasaki, and Siang Meng Sim. “Invariant Subspace Attack Against Midori64 and The Resistance Criteria for S-box Designs”. In: *IACR Transactions on Symmetric Cryptology* 2016.1 (2016), pp. 33–56. ISSN: 2519-173X (cit. on pp. 109, 117, 118, 363).

[Gur23] Gurobi Optimization, LLC. *Gurobi Optimizer Reference Manual*. 2023 (cit. on p. 274).

[Hao+20] Yonglin Hao, Gregor Leander, Willi Meier, Yosuke Todo, and Qingju Wang. “Modeling for Three-Subset Division Property Without Unknown Subset - Improved Cube Attacks Against Trivium and Grain-128AEAD”. In: *EUROCRYPT 2020, Part I*. Ed. by Anne Canteaut and Yuval Ishai. Vol. 12105. LNCS. Zagreb, Croatia: Springer, Cham, Switzerland, 2020, pp. 466–495 (cit. on p. 72).

[Hao+21] Yonglin Hao, Gregor Leander, Willi Meier, Yosuke Todo, and Qingju Wang. “Modeling for Three-Subset Division Property without Unknown Subset”. In: *Journal of Cryptology* 34.3 (July 2021), p. 22 (cit. on p. 72).

[Heb+20] Phil Hebborn, Baptiste Lambin, Gregor Leander, and Yosuke Todo. “Lower Bounds on the Degree of Block Ciphers”. In: *ASIACRYPT 2020, Part I*. Ed. by Shiho Moriai and Huaxiong Wang. Vol. 12491. LNCS. Daejeon, South Korea: Springer, Cham, Switzerland, 2020, pp. 537–566 (cit. on p. 72).

[Hei24] Rachelle Heim. “Symmetric cryptanalysis: from primitives to modes”. PhD thesis. Université Paris-Saclay, Oct. 2024 (cit. on p. 36).

[Her75] Israel Nathan Herstein. *Topics in algebra*. John Wiley & Sons, 1975 (cit. on p. 206).

[HLU23] Phil Hebborn, Gregor Leander, and Aleksei Udovenko. “Mathematical aspects of division property”. In: *Cryptography and Communications* 15.4 (2023), pp. 731–774 (cit. on p. 72).

[HMS23] Clément Hoffmann, Pierrick Méaux, and François-Xavier Standaert. “The Patching Landscape of Elisabeth-4 and the Mixed Filter Permutator Paradigm”. In: *INDOCRYPT 2023, Part I*. Ed. by Anupam Chattopadhyay, Shivam Bhasin, Stjepan Picek, and Chester Rebeiro. Vol. 14459. LNCS. Goa, India: Springer, Cham, Switzerland, 2023, pp. 134–156 (cit. on p. 282).

[HN12] Miia Hermelin and Kaisa Nyberg. “Multidimensional linear distinguishing attacks and Boolean functions”. In: *Cryptography and Communications* 4.1 (2012), pp. 47–64 (cit. on p. 295).

[Hou06] Xiang-dong Hou. “Affinity of permutations of $\mathbb{F}_2^n$”. In: *Discret. Appl. Math.* 154.2 (2006), pp. 313–325 (cit. on p. 166).

[HPS22] Sartaj Ul Hasan, Mohit Pal, and Pantelimon Stănică. “The c-Differential Uniformity and Boomerang Uniformity of Two Classes of Permutation Polynomials”. In: *IEEE Transactions on Information Theory* 68.1 (2022), pp. 679–691 (cit. on p. 160).

[Hu+20] Kai Hu, Siwei Sun, Meiqin Wang, and Qingju Wang. “An Algebraic Formulation of the Division Property: Revisiting Degree Evaluations, Cube Attacks, and Key-Independent Sums”. In: *ASIACRYPT 2020, Part I*. Ed. by Shiho Moriai and Huaxiong Wang. Vol. 12491. LNCS. Daejeon, South Korea: Springer, Cham, Switzerland, 2020, pp. 446–476 (cit. on p. 72).

[Hu+23] Kai Hu, Thomas Peyrin, Quan Quan Tan, and Trevor Yap. “Revisiting Higher-Order Differential-Linear Attacks from an Algebraic Perspective”. In: *ASIACRYPT 2023, Part III*. Ed. by Jian Guo and Ron Steinfeld. Vol. 14440. LNCS. Guangzhou, China: Springer, Singapore, Singapore, 2023, pp. 405–435 (cit. on pp. 67, 68, 79, 80, 82, 107).

[Hu24] Kai Hu. “Improved Conditional Cube Attacks on Ascon AEADs in Nonce-Respecting Settings”. In: *IACR Transactions on Symmetric Cryptology* 2024.2 (2024), 118–140 (cit. on pp. 80, 81).

[Hua+17] Senyang Huang, Xiaoyun Wang, Guangwu Xu, Meiqin Wang, and Jingyuan Zhao. “Conditional Cube Attack on Reduced-Round Keccak Sponge Function”. In: *EUROCRYPT 2017, Part II*. Ed. by Jean-Sébastien Coron and Jesper Buus Nielsen. Vol. 10211. LNCS. Paris, France: Springer, Cham, Switzerland, 2017, pp. 259–288 (cit. on pp. 74, 81).

[Ind+08] Sebastiaan Indesteege, Elena Andreeva, Christophe De Cannière, Orr Dunkelman, Emilia Käsper, Svetla Nikova, Bart Preneel, and Elmar Tischhause. *The Lane hash function.* Submission to NIST SHA-3 Cryptographic Hash Algorithm Competition. Available at `https://www.cosic.esat.kuleuven.be/lane/index.shtml`. 2008 (cit. on p. 260).

[Int24] Intel. *Optimizing Earlier Generations of Intel® 64 and IA-32 Processor Architectures, Throughput, and Latency.* 2024 (cit. on pp. 263, 264).

[Iso+23] Takanori Isobe, Ryoma Ito, Fukang Liu, Kazuhiko Minematsu, Motoki Nakahashi, Kosei Sakamoto, and Rentaro Shiba. "Areion: Highly-Efficient Permutations and Its Applications to Hash Functions for Short Input". In: *IACR TCHES* 2023.2 (2023), pp. 115–154 (cit. on p. 260).

[Jea16] Jérémy Jean. "Cryptanalysis of Haraka". In: *IACR Transactions on Symmetric Cryptology* 2016.1 (2016), pp. 1–12. ISSN: 2519-173X (cit. on p. 109).

[Jea+21] Jérémy Jean, Ivica Nikolic, Thomas Peyrin, and Yannick Seurin. "The Deoxys AEAD Family". In: *Journal of Cryptology* 34.3 (July 2021), p. 31 (cit. on pp. 260, 275, 366).

[Jeo+22] Jaeseong Jeong, Chang Heon Kim, Namhun Koo, Soonhak Kwon, and Sumin Lee. "On Cryptographic Parameters of Permutation Polynomials of the form $x^r h(x^{(2^n-1)/d})$". In: *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.* 105-A.8 (2022), pp. 1134–1146 (cit. on p. 228).

[JK97] Thomas Jakobsen and Lars R. Knudsen. "The Interpolation Attack on Block Ciphers". In: *FSE'97.* Ed. by Eli Biham. Vol. 1267. LNCS. Haifa, Israel: Springer, Berlin, Heidelberg, Germany, 1997, pp. 28–40 (cit. on p. 34).

[JN16] Jérémy Jean and Ivica Nikolic. "Efficient Design Strategies Based on the AES Round Function". In: *FSE 2016.* Ed. by Thomas Peyrin. Vol. 9783. LNCS. Bochum, Germany: Springer, Berlin, Heidelberg, Germany, 2016, pp. 334–353 (cit. on pp. 260, 263, 265).

[JW93] Heeralal Janwa and Richard M. Wilson. "Hyperplane Sections of Fermat Varieties in P^3 in Char.2 and Some Applications to Cyclic Codes". In: *Applied Algebra, Algebraic Algorithms and Error-Correcting Codes - AAECC-10.* Ed. by Gérard D. Cohen, Teo Mora, and Oscar Moreno. Vol. 673. Lecture Notes in Computer Science. Springer, 1993, pp. 180–194 (cit. on p. 189).

[Kah96] David Kahn. *The Codebreakers: The Comprehensive History of Secret Communication from Ancient Times to the Internet.* Scribner, 1996. ISBN: 9781439103555 (cit. on p. 1).

[Kas71] Tadao Kasami. “The Weight Enumerators for Several Classes of Subcodes of the 2nd Order Binary Reed-Muller Codes”. In: *Inf. Control.* 18.4 (1971), pp. 369–394 (cit. on pp. 189, 365).

[Ker83] Auguste Kerckhoffs. “La cryptographie militaire”. In: *Journal des Sciences Militaires* 9 (1883), pp. 5–38,161–191 (cit. on p. 2).

[KKK23] Lukas Kölsch, Björn Kriepke, and Gohar M. Kyureghyan. “Image sets of perfectly nonlinear maps”. In: *Designs, Codes and Cryptography* 91.1 (2023), pp. 1–27 (cit. on pp. 190, 191, 195–197, 217, 220, 238, 239).

[KN10] Dmitry Khovratovich and Ivica Nikolic. “Rotational Cryptanalysis of ARX”. In: *FSE 2010.* Ed. by Seokhie Hong and Tetsu Iwata. Vol. 6147. LNCS. Seoul, Korea: Springer, Berlin, Heidelberg, Germany, 2010, pp. 333–346 (cit. on p. 159).

[Knu93] Lars R. Knudsen. “Iterative Characteristics of DES and s^2-DES”. In: *CRYPTO’92.* Ed. by Ernest F. Brickell. Vol. 740. LNCS. Santa Barbara, CA, USA: Springer, Berlin, Heidelberg, Germany, 1993, pp. 497–511 (cit. on p. 44).

[Knu95] Lars R. Knudsen. “Truncated and Higher Order Differentials”. In: *FSE’94.* Ed. by Bart Preneel. Vol. 1008. LNCS. Leuven, Belgium: Springer, Berlin, Heidelberg, Germany, 1995, pp. 196–211 (cit. on pp. 63, 69, 361).

[Köl+16] Stefan Kölbl, Martin M. Lauridsen, Florian Mendel, and Christian Rechberger. *Haraka - Efficient Short-Input Hashing for Post-Quantum Applications.* Cryptology ePrint Archive, Report 2016/098. 2016 (cit. on p. 260).

[Kol98] Andrei N. Kolmogorov. “On Tables of Random Numbers (Reprinted from "Sankhya: The Indian Journal of Statistics", Series A, Vol. 25 Part 4, 1963)”. In: *Theor. Comput. Sci.* 207.2 (1998), pp. 387–395 (cit. on p. 15).

[KP02] Sergei Konyagin and Francesco Pappalardi. “Enumerating Permutation Polynomials over Finite Fields by Degree”. In: *Finite Fields and Their Applications* 8.4 (2002), pp. 548–553. ISSN: 1071-5797 (cit. on pp. 32, 35).

[KR21] Ted Krovetz and Phillip Rogaway. “The Design and Evolution of OCB”. In: *Journal of Cryptology* 34.4 (Oct. 2021), p. 36 (cit. on pp. 257, 260).

[KS07] Liam Keliher and Jiayuan Sui. “Exact maximum expected differential and linear probability for two-round Advanced Encryption Standard”. In: *IET Inf. Secur.* 1.2 (2007), pp. 53–57 (cit. on p. 262).

[KS99] Aviad Kipnis and Adi Shamir. "Cryptanalysis of the HFE Public Key Cryptosystem by Relinearization". In: *CRYPTO'99*. Ed. by Michael J. Wiener. Vol. 1666. LNCS. Santa Barbara, CA, USA: Springer, Berlin, Heidelberg, Germany, 1999, pp. 19–30 (cit. on p. 31).

[KW02] Lars R. Knudsen and David Wagner. "Integral Cryptanalysis". In: *FSE 2002*. Ed. by Joan Daemen and Vincent Rijmen. Vol. 2365. LNCS. Leuven, Belgium: Springer, Berlin, Heidelberg, Germany, 2002, pp. 112–127 (cit. on pp. 63, 69, 361).

[KZ21] Christian Kaspers and Yue Zhou. "The Number of Almost Perfect Nonlinear Functions Grows Exponentially". In: *J. Cryptol.* 34.1 (2021), p. 4 (cit. on pp. 218, 254).

[KZ22] Christian Kaspers and Yue Zhou. "A lower bound on the number of inequivalent APN functions". In: *J. Comb. Theory A* 186 (2022), p. 105554 (cit. on pp. 254, 255).

[Lai07] Yann Laigle-Chapuy. "Permutation polynomials and applications to coding theory". In: *Finite Fields and Their Applications* 13.1 (2007), pp. 58–70 (cit. on p. 198).

[Lai94] Xuejia Lai. "Higher Order Derivatives and Differential Cryptanalysis". In: *Communications and Cryptography: Two Sides of One Tapestry*. Ed. by Richard E. Blahut, Daniel J. Costello, Ueli Maurer, and Thomas Mittelholzer. Boston, MA: Springer US, 1994, pp. 227–233. ISBN: 978-1-4615-2694-0 (cit. on pp. 63, 65, 66, 361).

[Lai95] Xuejia Lai. "Additive and Linear Structures of Cryptographic Functions". In: *FSE'94*. Ed. by Bart Preneel. Vol. 1008. LNCS. Leuven, Belgium: Springer, Berlin, Heidelberg, Germany, 1995, pp. 75–85 (cit. on pp. 144, 145).

[LDW17] Zheng Li, Xiaoyang Dong, and Xiaoyun Wang. "Conditional Cube Attack on Round-Reduced ASCON". In: *IACR Transactions on Symmetric Cryptology* 2017.1 (2017), pp. 175–202. ISSN: 2519-173X (cit. on pp. 80, 81, 92, 103).

[Lea+11] Gregor Leander, Mohamed Ahmed Abdelraheem, Hoda AlKhzaimi, and Erik Zenner. "A Cryptanalysis of PRINTcipher: The Invariant Subspace Attack". In: *CRYPTO 2011*. Ed. by Phillip Rogaway. Vol. 6841. LNCS. Santa Barbara, CA, USA: Springer, Berlin, Heidelberg, Germany, 2011, pp. 206–221 (cit. on p. 117).

[Li+17] Yanbin Li, Guoyan Zhang, Wei Wang, and Meiqin Wang. "Cryptanalysis of round-reduced ASCON". In: *Sci. China Inf. Sci.* 60.3 (2017), p. 38102 (cit. on p. 80).

[Li+18] Yongqiang Li, Shizhu Tian, Yuyin Yu, and Mingsheng Wang. "On the Generalization of Butterfly Structure". In: *IACR Transactions on Symmetric Cryptology* 2018.1 (2018), pp. 160–179. ISSN: 2519-173X (cit. on p. 200).

[Li+21] Kangquan Li, Chunlei Li, Tor Helleseth, and Longjiang Qu. "A Complete Characterization of the APN Property of a Class of Quadrinomials". In: *IEEE Transactions on Information Theory* 67.11 (2021), pp. 7535–7549 (cit. on pp. 198, 246).

[Li+22] Kangquan Li, Yue Zhou, Chunlei Li, and Longjiang Qu. "Two new families of quadratic APN functions". In: *IEEE Transactions on Information Theory* 68.7 (2022), pp. 4761–4769 (cit. on pp. 218, 344, 347, 349).

[LK23] Kangquan Li and Nikolay Kaleyski. "Two new infinite families of APN functions in trivariate form". In: *IEEE Transactions on Information Theory* (2023) (cit. on pp. 216, 344, 345, 349).

[LMM91] Xuejia Lai, James L. Massey, and Sean Murphy. "Markov Ciphers and Differential Cryptanalysis". In: *EUROCRYPT'91*. Ed. by Donald W. Davies. Vol. 547. LNCS. Brighton, UK: Springer, Berlin, Heidelberg, Germany, 1991, pp. 17–38 (cit. on pp. 40, 41, 43, 44).

[LMR15] Gregor Leander, Brice Minaud, and Sondre Rønjom. "A Generic Approach to Invariant Subspace Attacks: Cryptanalysis of Robin, iSCREAM and Zorro". In: *EUROCRYPT 2015, Part I*. Ed. by Elisabeth Oswald and Marc Fischlin. Vol. 9056. LNCS. Sofia, Bulgaria: Springer, Berlin, Heidelberg, Germany, 2015, pp. 254–283 (cit. on pp. 109, 117, 155, 160, 161, 166, 174, 364).

[LN96] Rudolf Lidl and Harald Niederreiter. *Finite Fields*. 2nd ed. Encyclopedia of Mathematics and its Applications. Cambridge University Press, 1996 (cit. on pp. 19, 21, 51, 52, 211).

[Lou+12] Yu Lou, Huiting Han, Chunming Tang, and Maozhi Xu. *Constructing Vectorial Boolean Functions with High Algebraic Immunity Based on Group Decomposition*. Cryptology ePrint Archive, Paper 2012/335. `https://eprint.iacr.org/2012/335`. 2012 (cit. on p. 357).

[LP20] Gaëtan Leurent and Clara Pernot. *New Representations of the AES Key Schedule*. Cryptology ePrint Archive, Report 2020/1253. 2020 (cit. on p. 12).

[LP21] Gaëtan Leurent and Clara Pernot. "New Representations of the AES Key Schedule". In: *EUROCRYPT 2021, Part I*. Ed. by Anne Canteaut and François-Xavier Standaert. Vol. 12696. LNCS. Zagreb, Croatia: Springer, Cham, Switzerland, 2021, pp. 54–84 (cit. on pp. 117, 128).

[LPS21] Gaëtan Leurent, Clara Pernot, and André Schrottenloher. "Clustering Effect in Simon and Simeck". In: *ASIACRYPT 2021, Part I*. Ed. by Mehdi Tibouchi and Huaxiong Wang. Vol. 13090. LNCS. Singapore: Springer, Cham, Switzerland, 2021, pp. 272–302 (cit. on p. 45).

[Luc02] Stefan Lucks. “The Saturation Attack - A Bait for Twofish”. In: *FSE 2001.* Ed. by Mitsuru Matsui. Vol. 2355. LNCS. Yokohama, Japan: Springer, Berlin, Heidelberg, Germany, 2002, pp. 1–15 (cit. on p. 69).

[LW17] Li Lin and Wenling Wu. “Meet-in-the-Middle Attacks on Reduced-Round Midori64”. In: *IACR Transactions on Symmetric Cryptology* 2017.1 (2017), pp. 215–239. ISSN: 2519-173X (cit. on p. 109).

[Mat94] Mitsuru Matsui. “Linear Cryptanalysis Method for DES Cipher”. In: *EUROCRYPT’93.* Ed. by Tor Helleseth. Vol. 765. LNCS. Lofthus, Norway: Springer, Berlin, Heidelberg, Germany, 1994, pp. 386–397 (cit. on pp. 16, 46, 360).

[McE78] Robert J. McEliece. *A public-key cryptosystem based on algebraic coding theory.* The Deep Space Network Progress Report 42-44. https://ipnpr.jpl.nasa.gov/progress_report2/42-44/44N.PDF. Jet Propulsion Laboratory, California Institute of Technology, 1978, pp. 114–116 (cit. on pp. 2, 3).

[McF73] Robert L McFarland. “A family of difference sets in non-cyclic groups”. In: *Journal of Combinatorial Theory, Series A* 15.1 (1973), pp. 1–10 (cit. on p. 56).

[Méa+16] Pierrick Méaux, Anthony Journault, François-Xavier Standaert, and Claude Carlet. “Towards Stream Ciphers for Efficient FHE with Low-Noise Ciphertexts”. In: *EUROCRYPT 2016, Part I.* Ed. by Marc Fischlin and Jean-Sébastien Coron. Vol. 9665. LNCS. Vienna, Austria: Springer, Berlin, Heidelberg, Germany, 2016, pp. 311–343 (cit. on pp. 258, 282, 286).

[Méa+19] Pierrick Méaux, Claude Carlet, Anthony Journault, and François-Xavier Standaert. “Improved Filter Permutators for Efficient FHE: Better Instances and Implementations”. In: *INDOCRYPT 2019.* Ed. by Feng Hao, Sushmita Ruj, and Sourav Sen Gupta. Vol. 11898. LNCS. Hyderabad, India: Springer, Cham, Switzerland, 2019, pp. 68–91 (cit. on pp. 282, 299).

[Mes+21] Sihem Mesnager, Constanza Riera, Pantelimon Stănică, Haode Yan, and Zhengchun Zhou. “Investigations on c-(Almost) Perfect Nonlinear Functions”. In: *IEEE Transactions on Information Theory* 67.10 (2021), pp. 6916–6925 (cit. on p. 160).

[Min16] Brice Minaud. “Analyse de primitives cryptographiques récentes”. PhD thesis. Université de Rennes 1, 2016 (cit. on p. 161).

[MN17] Bart Mennink and Samuel Neves. “Encrypted Davies-Meyer and Its Dual: Towards Optimal Security Using Mirror Theory”. In: *CRYPTO 2017, Part III.* Ed. by Jonathan Katz and Hovav Shacham. Vol. 10403. LNCS. Santa Barbara, CA, USA: Springer, Cham, Switzerland, 2017, pp. 556–583 (cit. on pp. 262, 279).

[Mou+11] Nicky Mouha, Qingju Wang, Dawu Gu, and Bart Preneel. "Differential and Linear Cryptanalysis Using Mixed-Integer Linear Programming". In: *Information Security and Cryptology - 7th International Conference, Inscrypt 2011, Beijing, China, November 30 - December 3, 2011. Revised Selected Papers*. Ed. by Chuankun Wu, Moti Yung, and Dongdai Lin. Vol. 7537. Lecture Notes in Computer Science. Springer, 2011, pp. 57–76 (cit. on pp. 275, 297).

[MP13] Gary L. Mullen and Daniel Panario, eds. *Handbook of Finite Fields*. CRC Press, 2013 (cit. on p. 206).

[MS88] Willi Meier and Othmar Staffelbach. "Fast Correlation Attacks on Stream Ciphers (Extended Abstract)". In: *EUROCRYPT'88*. Ed. by C. G. Günther. Vol. 330. LNCS. Davos, Switzerland: Springer, Berlin, Heidelberg, Germany, 1988, pp. 301–314 (cit. on pp. 289, 295).

[MS89] Willi Meier and Othmar Staffelbach. "Fast Correlation Attacks on Certain Stream Ciphers". In: *Journal of Cryptology* 1.3 (Oct. 1989), pp. 159–176 (cit. on p. 14).

[MT06] Kazuhiko Minematsu and Yukiyasu Tsunoo. "Provably Secure MACs from Differentially-Uniform Permutations and AES-Based Implementations". In: *FSE 2006*. Ed. by Matthew J. B. Robshaw. Vol. 4047. LNCS. Graz, Austria: Springer, Berlin, Heidelberg, Germany, 2006, pp. 226–241 (cit. on p. 262).

[MV04] David A. McGrew and John Viega. *The Galois/Counter Mode of Operation (GCM)*. Submission to NIST Modes of Operation Process. Available at https://csrc.nist.rip/groups/ST/toolkit/BCM/documents/proposedmodes/gcm/gcm-spec.pdf. Jan. 2004 (cit. on pp. 13, 257, 260, 262).

[Nes00] *NESSIE - New European Schemes for Signatures, Integrity, and Encryption*. https://web.archive.org/web/20050305130550/http://www.cosic.esat.kuleuven.ac.be/nessie/. Mar. 2000 (cit. on p. 4).

[NFI24] Yuto Nakano, Kazuhide Fukushima, and Takanori Isobe. *Encryption algorithm Rocca-S*. Internet-Draft draft-nakano-rocca-s-05. Work in Progress. Internet Engineering Task Force, Jan. 2024. 25 pp. (cit. on pp. 260, 263).

[Nih72] Yoji Niho. "Multi-valued cross-correlation functions between two maximal linear recursive sequences on comma-free codes". https://apps.dtic.mil/sti/citations/AD0786066. PhD thesis. University of Southern California, 1972 (cit. on p. 189).

[Nik14] Ivica Nikolić. *Tiaoxin–346*. Submission to CAESAR Competition. Available at https://competitions.cr.yp.to/round3/tiaoxinv21.pdf. 2014 (cit. on pp. 260, 263, 366).

[Nik17] Ivica Nikolic. “How to Use Metaheuristics for Design of Symmetric-Key Primitives”. In: *ASIACRYPT 2017, Part III*. Ed. by Tsuyoshi Takagi and Thomas Peyrin. Vol. 10626. LNCS. Hong Kong, China: Springer, Cham, Switzerland, 2017, pp. 369–391 (cit. on pp. 260, 263).

[Nis07] *Hash Functions*. https://csrc.nist.gov/Projects/hash-functions/sha-3-project. Jan. 2007 (cit. on pp. 62, 78).

[Nis17] *NIST Lightweight Cryptography Competition*. https://csrc.nist.gov/Projects/lightweight-cryptography. Jan. 2017 (cit. on pp. 2, 4, 63, 360).

[Nis22] *NIST Post-Quantum Cryptography Standardization - Round 4 Submissions*. https://csrc.nist.gov/Projects/post-quantum-cryptography/round-4-submissions. July 2022 (cit. on p. 3).

[NK93] Kaisa Nyberg and Lars R. Knudsen. “Provable Security Against Differential Cryptanalysis (Rump Session)”. In: *CRYPTO'92*. Ed. by Ernest F. Brickell. Vol. 740. LNCS. Santa Barbara, CA, USA: Springer, Berlin, Heidelberg, Germany, 1993, pp. 566–574 (cit. on pp. 37, 189, 190, 361, 365).

[NLV11] Michael Naehrig, Kristin E. Lauter, and Vinod Vaikuntanathan. “Can homomorphic encryption be practical?” In: *Proceedings of the 3rd ACM Cloud Computing Security Workshop, CCSW 2011, Chicago, IL, USA, October 21, 2011*. Ed. by Christian Cachin and Thomas Ristenpart. ACM, 2011, pp. 113–124 (cit. on pp. 258, 280).

[NW05] Harald Niederreiter and Arne Winterhof. “Cyclotomic $\mathcal{R}$-orthomorphisms of finite fields”. In: *Discret. Math.* 295.1-3 (2005), pp. 161–171 (cit. on p. 196).

[Nyb91] Kaisa Nyberg. “Perfect Nonlinear S-Boxes”. In: *EUROCRYPT'91*. Ed. by Donald W. Davies. Vol. 547. LNCS. Brighton, UK: Springer, Berlin, Heidelberg, Germany, 1991, pp. 378–386 (cit. on pp. 38, 55).

[Nyb94] Kaisa Nyberg. “Differentially Uniform Mappings for Cryptography”. In: *EUROCRYPT'93*. Ed. by Tor Helleseth. Vol. 765. LNCS. Lofthus, Norway: Springer, Berlin, Heidelberg, Germany, 1994, pp. 55–64 (cit. on pp. 35, 37, 189, 190, 365).

[Nyb95] Kaisa Nyberg. “Linear Approximation of Block Ciphers (Rump Session)”. In: *EUROCRYPT'94*. Ed. by Alfredo De Santis. Vol. 950. LNCS. Perugia, Italy: Springer, Berlin, Heidelberg, Germany, 1995, pp. 439–444 (cit. on pp. 48, 51).

[Per19] Léo Perrin. “Partitions in the S-Box of Streebog and Kuznyechik”. In: *IACR Transactions on Symmetric Cryptology* 2019.1 (2019), pp. 302–329. ISSN: 2519-173X (cit. on pp. 255, 353, 354, 366).

[PT22] Thomas Peyrin and Quan Quan Tan. “Mind Your Path: On (Key) Dependencies in Differential Characteristics”. In: *IACR Transactions on Symmetric Cryptology* 2022.4 (2022), pp. 179–207 (cit. on p. 44).

[PU16] Léo Perrin and Aleksei Udovenko. “Exponential S-Boxes: a Link Between the S-Boxes of BelT and Kuznyechik/Streebog”. In: *IACR Transactions on Symmetric Cryptology* 2016.2 (2016), pp. 99–124. ISSN: 2519-173X (cit. on pp. 255, 353, 366).

[PUB16] Léo Perrin, Aleksei Udovenko, and Alex Biryukov. “Cryptanalysis of a Theorem: Decomposing the Only Known Solution to the Big APN Problem”. In: *CRYPTO 2016, Part II*. Ed. by Matthew Robshaw and Jonathan Katz. Vol. 9815. LNCS. Santa Barbara, CA, USA: Springer, Berlin, Heidelberg, Germany, 2016, pp. 93–122 (cit. on pp. 190, 200).

[RBB03] Phillip Rogaway, Mihir Bellare, and John Black. “OCB: A block-cipher mode of operation for efficient authenticated encryption”. In: *ACM Trans. Inf. Syst. Secur.* 6.3 (2003), 365–403. ISSN: 1094-9224 (cit. on pp. 13, 257, 260).

[Reg05] Oded Regev. “On lattices, learning with errors, random linear codes, and cryptography”. In: *STOC 2005*. Ed. by Harold N. Gabow and Ronald Fagin. ACM, 2005, pp. 84–93 (cit. on pp. 280, 367).

[Rij+02] Vincent Rijmen, Bart Van Rompay, Bart Preneel, and Joos Vandewalle. “Producing Collisions for PANAMA”. In: *FSE 2001*. Ed. by Mitsuru Matsui. Vol. 2355. LNCS. Yokohama, Japan: Springer, Berlin, Heidelberg, Germany, 2002, pp. 37–51 (cit. on p. 265).

[Rog02] Phillip Rogaway. “Authenticated-Encryption With Associated-Data”. In: *ACM CCS 2002*. Ed. by Vijayalakshmi Atluri. Washington, DC, USA: ACM Press, 2002, pp. 98–107 (cit. on pp. 7, 75, 361).

[Roh+21] Raghvendra Rohit, Kai Hu, Sumanta Sarkar, and Siwei Sun. “Misuse-Free Key-Recovery and Distinguishing Attacks on 7-Round Ascon”. In: *IACR Transactions on Symmetric Cryptology* 2021.1 (2021), pp. 130–155. ISSN: 2519-173X (cit. on pp. 79–81, 98, 103).

[Røn16] Sondre Rønjom. *Invariant subspaces in Simpira*. Cryptology ePrint Archive, Report 2016/248. 2016 (cit. on p. 109).

[RP20] Adrián Ranea and Bart Preneel. “On Self-equivalence Encodings in White-Box Implementations”. In: *SAC 2020*. Ed. by Orr Dunkelman, Michael J. Jacobson Jr., and Colin O’Flynn. Vol. 12804. LNCS. Halifax, NS, Canada (Virtual Event): Springer, Cham, Switzerland, 2020, pp. 639–669 (cit. on p. 167).

[RS21] Raghvendra Rohit and Santanu Sarkar. “Diving Deep into the Weak Keys of Round Reduced Ascon”. In: *IACR Transactions on Symmetric Cryptology* 2021.4 (2021), pp. 74–99 (cit. on pp. 79–81, 103).

[RSA78] Ronald L. Rivest, Adi Shamir, and Leonard M. Adleman. “A Method for Obtaining Digital Signatures and Public-Key Cryptosystems”. In: *Communications of the Association for Computing Machinery* 21.2 (Feb. 1978), pp. 120–126 (cit. on pp. 2, 3).

[Sage24] The Sage Developers. *SageMath, the Sage Mathematics Software System (Version 9.7)*. https://www.sagemath.org. 2024 (cit. on pp. 104, 254).

[Sak+21] Kosei Sakamoto, Fukang Liu, Yuto Nakano, Shinsaku Kiyomoto, and Takanori Isobe. "Rocca: An Efficient AES-based Encryption Scheme for Beyond 5G". In: *IACR Transactions on Symmetric Cryptology* 2021.2 (2021), pp. 1–30. ISSN: 2519-173X (cit. on pp. 260, 263, 265, 275).

[Sak+22] Kosei Sakamoto, Fukang Liu, Yuto Nakano, Shinsaku Kiyomoto, and Takanori Isobe. *Rocca: An Efficient AES-based Encryption Scheme for Beyond 5G (Full version)*. Cryptology ePrint Archive, Report 2022/116. 2022 (cit. on pp. 260, 263).

[Sha3] *SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions*. National Institute of Standards and Technology, NIST FIPS PUB 202, U.S. Department of Commerce. Aug. 2015 (cit. on p. 286).

[Sha49] Claude E. Shannon. "Communication theory of secrecy systems". In: *Bell Systems Technical Journal* 28.4 (1949), pp. 656–715 (cit. on pp. 2, 5, 10, 29, 31).

[Sho96] Victor Shoup. "On Fast and Provably Secure Message Authentication Based on Universal Hashing". In: *CRYPTO'96*. Ed. by Neal Koblitz. Vol. 1109. LNCS. Santa Barbara, CA, USA: Springer, Berlin, Heidelberg, Germany, 1996, pp. 313–328 (cit. on p. 262).

[Sie84] Thomas Siegenthaler. "Correlation-immunity of nonlinear combining functions for cryptographic applications". In: *IEEE Transactions on Information Theory* 30.5 (1984), pp. 776–780 (cit. on p. 14).

[Sie85] Thomas Siegenthaler. "Decrypting a Class of Stream Ciphers Using Ciphertext Only". In: *IEEE Trans. Computers* 34.1 (1985), pp. 81–85 (cit. on pp. 14, 289).

[Sin64] Richard C. Singleton. "Maximum distance q -nary codes". In: *IEEE Transactions on Information Theory* 10.2 (1964), pp. 116–118 (cit. on p. 290).

[SNC09] Mate Soos, Karsten Nohl, and Claude Castelluccia. "Extending SAT Solvers to Cryptographic Problems". In: *Theory and Applications of Satisfiability Testing - SAT 2009*. Ed. by Oliver Kullmann. Vol. 5584. Lecture Notes in Computer Science. Springer, 2009, pp. 244–257 (cit. on p. 98).

[Stă+22] Pantelimon Stănică, Sugata Gangopadhyay, Aaron Geary, Constanza Riera, and Anton Tkachenko. "C-differential bent functions and perfect nonlinearity". In: *Discrete Applied Mathematics* 307 (2022), pp. 160–171. ISSN: 0166-218X (cit. on p. 160).

[Sti92] Douglas R. Stinson. “Universal Hashing and Authentication Codes”. In: *CRYPTO’91*. Ed. by Joan Feigenbaum. Vol. 576. LNCS. Santa Barbara, CA, USA: Springer, Berlin, Heidelberg, Germany, 1992, pp. 74–85 (cit. on p. 279).

[Tan19] Hiroaki Taniguchi. “On some quadratic APN functions”. In: *Designs, Codes and Cryptography* 87 (2019), pp. 1973–1983 (cit. on pp. 216, 344, 348).

[TG92] Anne Tardy-Corfdir and Henri Gilbert. “A Known Plaintext Attack of FEAL-4 and FEAL-6”. In: *CRYPTO’91*. Ed. by Joan Feigenbaum. Vol. 576. LNCS. Santa Barbara, CA, USA: Springer, Berlin, Heidelberg, Germany, 1992, pp. 172–181 (cit. on pp. 16, 46, 360).

[TLS16] Yosuke Todo, Gregor Leander, and Yu Sasaki. “Nonlinear Invariant Attack - Practical Attack on Full SCREAM, iSCREAM, and Midori64”. In: *ASIACRYPT 2016, Part II*. Ed. by Jung Hee Cheon and Tsuyoshi Takagi. Vol. 10032. LNCS. Hanoi, Vietnam: Springer, Berlin, Heidelberg, Germany, 2016, pp. 3–33 (cit. on p. 109).

[TLS19] Yosuke Todo, Gregor Leander, and Yu Sasaki. “Nonlinear Invariant Attack: Practical Attack on Full SCREAM, iSCREAM, and Midori64”. In: *Journal of Cryptology* 32.4 (Oct. 2019), pp. 1383–1422 (cit. on pp. 109, 115, 118, 121, 122, 125, 133, 174, 363).

[TM16] Yosuke Todo and Masakatu Morii. “Bit-Based Division Property and Application to Simon Family”. In: *FSE 2016*. Ed. by Thomas Peyrin. Vol. 9783. LNCS. Bochum, Germany: Springer, Berlin, Heidelberg, Germany, 2016, pp. 357–377 (cit. on p. 72).

[Tod15a] Yosuke Todo. “Integral Cryptanalysis on Full MISTY1”. In: *CRYPTO 2015, Part I*. Ed. by Rosario Gennaro and Matthew J. B. Robshaw. Vol. 9215. LNCS. Santa Barbara, CA, USA: Springer, Berlin, Heidelberg, Germany, 2015, pp. 413–432 (cit. on p. 72).

[Tod15b] Yosuke Todo. “Structural Evaluation by Generalized Integral Property”. In: *EUROCRYPT 2015, Part I*. Ed. by Elisabeth Oswald and Marc Fischlin. Vol. 9056. LNCS. Sofia, Bulgaria: Springer, Berlin, Heidelberg, Germany, 2015, pp. 287–314 (cit. on pp. 63, 71, 72).

[Tod17] Yosuke Todo. “Integral Cryptanalysis on Full MISTY1”. In: *Journal of Cryptology* 30.3 (July 2017), pp. 920–959 (cit. on p. 72).

[TSI23] Nobuyuki Takeuchi, Kosei Sakamoto, and Takanori Isobe. “On Optimality of the Round Function of Rocca”. In: *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.* 106.1 (2023), pp. 45–53 (cit. on p. 260).

[Vie07] Michael Vielhaber. *Breaking ONE.FIVIUM by AIDA an Algebraic IV Differential Attack*. Cryptology ePrint Archive, Report 2007/413. 2007 (cit. on pp. 14, 72).

[VV18] Serge Vaudenay and Damian Vizár. "Can Caesar Beat Galois? - Robustness of CAESAR Candidates Against Nonce Reusing and High Data Complexity Attacks". In: *ACNS 18 - International Conference on Applied Cryptography and Network Security*. Ed. by Bart Preneel and Frederik Vercauteren. Vol. 10892. LNCS. Leuven, Belgium: Springer, Cham, Switzerland, 2018, pp. 476–494 (cit. on pp. 84, 85).

[Wag04] David Wagner. "Towards a Unifying View of Block Cipher Cryptanalysis". In: *FSE 2004*. Ed. by Bimal K. Roy and Willi Meier. Vol. 3017. LNCS. New Delhi, India: Springer, Berlin, Heidelberg, Germany, 2004, pp. 16–33 (cit. on pp. 155, 158).

[Wan07] Qiang Wang. "Cyclotomic Mapping Permutation Polynomials over Finite Fields". In: *Sequences, Subsequences, and Consequences*. Ed. by Solomon W. Golomb, Guang Gong, Tor Helleseth, and Hong-Yeop Song. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 119–128. ISBN: 978-3-540-77404-4 (cit. on pp. 195, 197).

[Wan17] Qiang Wang. "A note on inverses of cyclotomic mapping permutation polynomials over finite fields". In: *Finite Fields and Their Applications* 45 (2017), pp. 422–427 (cit. on p. 198).

[Wel69] Charles Wells. "The degrees of permutation polynomials over finite fields". In: *Journal of Combinatorial Theory* 7.1 (1969), pp. 49–55 (cit. on p. 32).

[WL91] Daqing Wan and Rudolf Lidl. "Permutation polynomials of the form $x^r f(x^{(q-1)/d})$ and their group structure". In: *Monatshefte für Mathematik* 112 (1991), pp. 149–163 (cit. on p. 198).

[WP14] Hongjun Wu and Bart Preneel. "AEGIS: A Fast Authenticated Encryption Algorithm". In: *SAC 2013*. Ed. by Tanja Lange, Kristin Lauter, and Petr Lisonek. Vol. 8282. LNCS. Burnaby, BC, Canada: Springer, Berlin, Heidelberg, Germany, 2014, pp. 185–201 (cit. on pp. 260, 263, 366).

[Xia+16] Zejun Xiang, Wentao Zhang, Zhenzhen Bao, and Dongdai Lin. "Applying MILP Method to Searching Integral Distinguishers Based on Division Property for 6 Lightweight Block Ciphers". In: *ASIACRYPT 2016, Part I*. Ed. by Jung Hee Cheon and Tsuyoshi Takagi. Vol. 10031. LNCS. Hanoi, Vietnam: Springer, Berlin, Heidelberg, Germany, 2016, pp. 648–678 (cit. on p. 72).

[XM88] Guo-Zhen Xiao and James L. Massey. "A spectral characterization of correlation-immune combining functions". In: *IEEE Transactions on Information Theory* 34.3 (1988), pp. 569–571 (cit. on p. 293).

[Yan+15] Gangqiang Yang, Bo Zhu, Valentin Suder, Mark D. Aagaard, and Guang Gong. "The Simeck Family of Lightweight Block Ciphers". In: *CHES 2015*. Ed. by Tim Güneysu and Helena Handschuh. Vol. 9293. LNCS. Saint-Malo, France: Springer, Berlin, Heidelberg, Germany, 2015, pp. 307–329 (cit. on p. 10).

[YP22] Yuyin Yu and Léo Perrin. "Constructing more quadratic APN functions with the QAM method". In: *Cryptography and Communications* 14.6 (2022), pp. 1359–1369 (cit. on pp. 214, 218, 230, 232, 233).

[YT19] Chen-Dong Ye and Tian Tian. "Revisit Division Property Based Cube Attacks: Key-Recovery or Distinguishing Attacks?" In: *IACR Transactions on Symmetric Cryptology* 2019.3 (2019), pp. 81–102. ISSN: 2519-173X (cit. on p. 74).

[Yu+20] Yuyin Yu, Nikolay S. Kaleyski, Lilya Budaghyan, and Yongqiang Li. "Classification of quadratic APN functions with coefficients in $\mathbb{F}_2$ for dimensions up to 9". In: *Finite Fields and Their Applications* 68 (2020), p. 101733 (cit. on p. 223).

[YWL14] Yuyin Yu, Mingsheng Wang, and Yongqiang Li. "A matrix approach for constructing quadratic APN functions". In: *Designs, Codes and Cryptography* 73.2 (2014), pp. 587–600 (cit. on pp. 214, 218, 232, 233).

[Zhe+13] Jia Zheng, Baofeng Wu, Yufu Chen, and Zhuojun Liu. *Constructing $2m$-variable Boolean functions with optimal algebraic immunity based on polar decomposition of $\mathbb{F}^*_{2^{2m}}$*. arXiv 1304.2946. 2013 (cit. on p. 357).

[Zhe+22] Lijing Zheng, Haibin Kan, Yanjun Li, Jie Peng, and Deng Tang. "Constructing new APN functions through relative trace functions". In: *IEEE Transactions on Information Theory* 68.11 (2022), pp. 7528–7537 (cit. on pp. 215, 343, 347).

[ZP13] Yue Zhou and Alexander Pott. "A new family of semifields with 2 parameters". In: *Advances in Mathematics* 234 (2013), pp. 43–60 (cit. on pp. 216, 344, 348).

[ZZ99] Yuliang Zheng and Xian-Mo Zhang. "Plateaued Functions". In: *Information and Communication Security - ICICS'99*. Ed. by Vijay Varadharajan and Yi Mu. Vol. 1726. Lecture Notes in Computer Science. Springer, 1999, pp. 284–300 (cit. on p. 235).

List of Figures

List of Tables

Appendix A

Infinite families of quadratic APN functions

The polynomial representations of the known infinite families of APN functions, together with the conditions on their parameters to actually be APN are presented in Tables A.1 to A.4. In the following, we provide more details on these conditions when they do not fit in the tables, but also on the precise references where these results were found.

In the following, for $n = \ell k$, we denote by $S_{\ell,k} = \frac{2^n-1}{2^k-1} = \sum_{i=0}^{\ell-1} 2^{ik}$.

(BCL08a). See [BCL08, Corollary 1].

(BCL08b). See [BCL08, Theorem 2].

(BCV20). See [BCV20, Lemma 3.17].

(BCL09a). See [BCL09a, Corollary 1] [BCL09b, Corollaries 3 & 4].

(BBMM11). See [Bra+11a, Theorem 2.1].

(BCCCV20). See [Bud+20, Theorem VI.3 & Equation 16].

(BHK20). See [BHK20, Corollary 1].

(ZKLPT22). The conditions for this family of quadratic APN functions are numerous. We therefore recall the original statement by the authors.

Theorem A.1 (Family (ZKLPT22))**.** *[Zhe+22, Theorem 2] Let $n = 2k$ with $k \geq 1$ odd. Let i be a positive integer such that $\gcd(n, i) = 1$. Let $a \in \mathbb{F}_{2^n} \setminus \mathbb{F}_{2^k}$ and $b, c \in \mathbb{F}_{2^n}$ such that $bc \neq 0$. Let $F_{i,s,a,b,c}\colon x \to a\mathrm{Tr}_{\mathbb{L}/\mathbb{F}}(bx^{2^i+1}) + a^{2^k}\mathrm{Tr}_{\mathbb{L}/\mathbb{F}}(cx^{2^s+1})$. If one of the following conditions is verified, then $F_{i,s,a,b,c}$ is APN over $\mathbb{F}_{2^n}$:*

1. *b is not a cube and:*

   a) *$s = 3i$ and $\frac{c}{b^{2^{2i}-2^i+1}} \in \mathbb{F}_{2^k}^*$ or,*

   b) *$s = k - 2i$ and $\frac{c^{2^{2i}}}{b^{2^i-1}} \in \mathbb{F}_{2^k}^*$ or,* *c)* $s = k + 2i$ *and* $cb^{2^i-1} \in \mathbb{F}_{2^k}^*$ *or,*

   d) $i = 1$*,* $s = (k-2)^{-1} \bmod n$*, and* $\frac{c^{2^s-1}}{b^{2^{2s}}} \in \mathbb{F}_{2^k}^*$ *or,*

   e) $s = k$ *and* $c \notin \mathbb{F}_{2^k}$

2. *or,* $s = n - i$ *and* $\frac{c^{2^i}}{b} \notin \mathbb{F}_{2^k}$*.*

(LZLQ22a). See [Li+22, Theorem 6].

(LZLZ22b). See [Li+22, Theorem 5].

(ZP13). See [ZP13, Corrolary 2].

(T19). See [Tan19, Theorem 3].

(CBC21). See [CBC21, Theorem 6.2].

(G22a). See [Göl22, Theorem III.2 $\mathcal{F}_1$].

(G22b). See [Göl22, Theorem III.2 $\mathcal{F}_2$].

(GK21). See [GK21, Theorem 1].

(CLV22a). See [CLV22, Theorem 3].

(CLV22b). See [CLV22, Theorem 4].

(LK23a/b). The conditions for these two families of quadratic APN functions are too numerous for the table. We therefore recall the original statements by the authors.

Theorem A.2 (Family (LK23a) [LK23, Theorem 1])**.** *Let* $\gcd(s,k) = 1$ *and*

$$F\colon (x,y,z) \mapsto (x^{2^s+1} + x^{2^s}z + yz^{2^s}, x^{2^s}z + y^{2^s+1}, xy^{2^s} + y^{2^s}z + z^{2^s+1}).$$

Assume that the polynomials P_1, P_2, P_3 *have no root in* $\mathbb{F}_{2^k}$ *and* P_4 *have no root in* $\mathbb{F}_{2^k} \times \mathbb{F}_{2^k}$ *where* $q = 2^s$ *and* P_1, P_2, P_3, P_4 *are defined by:*

$$\begin{aligned} P_1 &= X^{q^2+q+1} + X + 1, \\ P_2 &= X^{q^2+q+1} + X^{q^2} + 1, \\ P_3 &= X^{q^2+q+1} + X^{q^2+1} + X^{q+1} + X + 1 \\ P_4 &= X^{q^2+q+1} + XY^{q^2+q} + XY^q + X^{q^2+q} + X^qY^{q^2} \\ &\quad + X^{q^2}Y + Y^{q^2+q+1} + Y^{q^2+q} + Y^{q^2} + Y^q + 1. \end{aligned}$$

Then F *is APN.*

Theorem A.3 (Family (LK23b) [LK23, Theorem 9])**.** *Let* $\gcd(s,k)=1$ *and*

$$F\colon (x,y,z)\mapsto (x^{2^s+1}+xy^{2^s}+yz^{2^s}, xy^{2^s}+z^{2^s+1}, x^{2^s}z+y^{2^s+1}+y^{2^s}z).$$

Assume that the polynomials P_1,P_2,P_3 *have no root in* $\mathbb{F}_{2^k}$ *and* P_4 *have no root in* $\mathbb{F}_{2^k}\times\mathbb{F}_{2^k}$ *where* $q=2^s$ *and* P_1,P_2,P_3,P_4 *are defined by:*

$$\begin{aligned}
P_1 &= X^{q^2+q+1}+X^{q+1}+X^q+X+1,\\
P_2 &= X^{q^2+q+1}+X^{q^2}+1,\\
P_3 &= X^{q^2+q+1}+X+1,\\
P_4 &= X^{q^2+q+1}+X^{q+1}Y^{q^2}+XY^q+X^qY^{q^2}+X^{q^2}Y\\
&\quad +X^{q^2}+Y^{q^2+q+1}+Y^{q^2+1}+Y^{q^2+q}+Y^{q^2}+1.
\end{aligned}$$

Then F *is APN.*

ID	Functions	Conditions to be APN	References
(BCL08a)	$x^{2^s+1} + ax^{2^{(3-i)k+s}+2^{ik}}$	**Field:** $n = 3k$ (*i.e.* $\ell = 3$), $k \geq 4$, $\gcd(3,k) = 1$ **Exponent:** $\gcd(s,n) = 1$ (Gold for $\mathbb{F}_{2^n}$) **Others:** $sk \equiv i \bmod 3, i \in \{1,2\}$, $\mathrm{ord}(a) = S_{3,k}$	[BCL08, Bud+06]
(BCL08b)	$x^{2^s+1} + ax^{2^{(4-i)k+s}+2^{ik}}$	**Field:** $n = 4k$ (*i.e.* $\ell = 4$), $k \geq 3$, k odd (so $\gcd(4,k) = 1$) **Exponent:** $\gcd(s,n) = 1$ (Gold for $\mathbb{F}_{2^n}$) **Others:** $sk \equiv i \bmod 4, i \in \{1,3\}$, $\mathrm{ord}(a) = S_{4,k}$	[BCL08, BCL06]
(BCV20)	$ax^{2^k+1} + x^{2^s+1} + x^{2^{s+k}+2^k} + bx^{2^{k+s}+1} + b^{2^k}x^{2^s+2^k}$	**Field:** $n = 2k$. **Exponent:** $\gcd(s,k) = 1$ (Gold for $\mathbb{F}_{2^k}$) **Others:** $a \in \mathbb{F}_{2^n} \setminus \mathbb{F}_{2^k}$, $X^{2^s+1} + bX^{2^s} + b^{2^k}X + 1$ has no root x s.t $x^{2^k+1} = 1$.	[BC08, BCV20]
(BCL09a)	$x^3 + a^{-1}\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_2}(a^3x^9)$	$a \neq 0$	[BCL09a]
(BCL09b/c)	$x^3 + a^{-1}\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^3}}(a^3x^9 + a^6x^{18})$ $x^3 + a^{-1}\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^3}}(a^6x^{18} + a^{12}x^{36})$	**Field:** $n = 3k$ **Exponent:** - **Others:** $a \neq 0$	[BCL09b]
(BBMM11)	$ax^{2^s+1} + a^{2^k}x^{2^{2k}+2^{k+s}} + bx^{2^{2k}+1} + ca^{2^k+1}x^{2^s+2^{k+s}}$	**Field:** $n = 3k$, $\gcd(3,k) = 1$. **Exponent:** $\gcd(s,n) = 1$ (Gold for $\mathbb{F}_{2^n}$), $3 \mid (k+s)$ **Others:** a primitive in $\mathbb{F}_{2^n}$. $b, c \in \mathbb{F}_{2^k}, bc \neq 1$.	[Bra+11a]

Table A.1: Known infinite families of univariate quadratic APN functions over $\mathbb{F}_{2^n}$ (1/2). The Gold mappings are omitted.

ID	Functions	Conditions to be APN	References
(BCCCV20)	$a^2x^{2^{2k+1}+1}+b^2x^{2^{k+1}+1}+ax^{2^{2k}+2}+bx^{2^k+2}+dx^3$	**Field:** $n=3k$. **Exponent:**. **Others:** see [Bud+20, Theorem VI.3].	[Bud+20]
(BHK20)	$x^3+ax^{2^{s+i}+2^i}+a^2x^{2^{k+1}+2^k}+x^{2^{s+i+k}+2^{i+k}}$	**Field:** $n=2k$, k odd, $3\nmid k$. **Exponent:** If i is even, $s\in\left\{k-2,\ k,\ n-1,\ (k-2)^{-1} \bmod n\right\}$. If i odd, $s\in\left\{k+2, n-1, (k+2)^{-1} \bmod n\right\}$. **Others:** a of order 3.	[BHK20]
(ZKLPT22)	$a\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(bx^{2^i+1})+a^{2^k}\mathrm{Tr}_{\mathbb{F}_{2^n}/\mathbb{F}_{2^k}}(cx^{2^s+1})$	**Field:** $n=2k$, k odd. **Exponent:** $\gcd(i,n)=1$ **Others:** $a\notin\mathbb{F}_{2^k}, bc\neq 0$. i,s,b,c satisfy the conditions of Theorem A.1.	[Zhe+22]
(LZLQ22a)	$L(x)^{2^k+1}+bx^{2^k+1}$	**Field:** $n=3k$. **Exponent:** $\gcd(s,k)=1$ (Gold for $\mathbb{F}_{2^k}$). **Others:** $a^{\frac{2^n-1}{2^k-1}}\neq 1$, $b\in\mathbb{F}_{2^k}^*$. $L\colon x\mapsto x^{2^{k+s}}+ax^{2^s}+x$ bijection over $\mathbb{F}_{2^n}$.	[Li+22]

Table A.2: Known infinite families of univariate quadratic APN functions over $\mathbb{F}_{2^n}$ (2/2).

ID	Functions	Conditions to be APN	References
(ZP13)	$(x,y) \mapsto \begin{pmatrix} x^{2^s+1} + ay^{(2^s+1)2^i} \\ xy \end{pmatrix}$	**Field:** $n = 2k$. k even. **Exponent:** $\gcd(s,k) = 1$ (Gold for $\mathbb{F}_{2^k}$), i even. **Others:** $a \in \mathbb{F}_{2^k}$ and non-cubic.	[ZP13]
(T19)	$(x,y) \mapsto \begin{pmatrix} x^{2^{2s}+2^{3s}} + ax^{2^{2s}}y^{2^s} + by^{2^s+1} \\ xy \end{pmatrix}$	**Field:** $n = 2k$. $k \geq 2$. **Exponent:** $\gcd(s,k) = 1$ (Gold for $\mathbb{F}_{2^k}$). **Others:** $a \in \mathbb{F}_{2^k}, b \in \mathbb{F}_{2^k}^*$ such that $X^{2^s+1} + aX + b$ has no root in $\mathbb{F}_{2^k}$.	[Tan19]
(CBC21)	$(x,y) \mapsto \begin{pmatrix} x^{2^s+1} + x^{2^{s+k/2}}y^{2^{k/2}} + axy^{2^s} + by^{2^s+1} \\ xy \end{pmatrix}$	**Field:** $n = 2k$. k even. **Exponent:** $\gcd(s,k) = 1$ (Gold for $\mathbb{F}_{2^k}$), $s < \frac{k}{2}$. **Others:** $a, b \in \mathbb{F}_{2^k}$ such that $(bX^{2^s+1} + aX^{2^s} + 1)^{2^{k/2}+1} + X^{2^{k/2}+1}$ has no root in $\mathbb{F}_{2^k}$.	[CBC21]
(G22a)	$(x,y) \mapsto \begin{pmatrix} x^{2^s+1} + xy^{2^s} + y^{2^s+1} \\ x^{2^{2s}+1} + x^{2^{2s}}y + y^{2^{2s}+1} \end{pmatrix}$	**Field:** $n = 2k$ **Exponent:** $\gcd(3s,k) = 1$.	[Göl22]
(G22b)	$(x,y) \mapsto \begin{pmatrix} x^{2^s+1} + xy^{2^s} + y^{2^s+1} \\ x^{2^{3s}}y + xy^{2^{3s}} \end{pmatrix}$	**Field:** $n = 2k$, k odd. **Exponent:** $\gcd(3s,k) = 1$.	[Göl22]
(GK21)	$(x,y) \mapsto \begin{pmatrix} x^{2^s+1} + by^{2^s+1} \\ x^{2^{s+k/2}}y + \frac{a}{b}xy^{2^{s+k/2}} \end{pmatrix}$	**Field:** $n = 2k$, $k \equiv 2 \bmod 4$. **Exponent:** $\gcd(s,k) = 1$. **Others:** $a \in \mathbb{F}_{2^{k/2}}^*$, $b \in \mathbb{F}_{2^k}$, b non-cubic such that $b^{2^s+2^{s+\frac{k}{2}}} \neq a^{2^s+1}$	[GK21]

Table A.3: Known infinite families of multivariate quadratic APN functions over $\mathbb{F}_{2^n}$ (1/2).

ID	Functions	Conditions to be APN	References
(LZLQ22b)	$(x,y) \mapsto \begin{pmatrix} x^3+xy^2+y^3+xy \\ x^5+x^4y+y^5+xy+x^2y^2 \end{pmatrix}$	**Field:** $n=2k$, $\gcd(k,3)=1$.	[Li+22]
(CLV22a)	$(x,y) \mapsto \begin{pmatrix} x^{2^s+1}+xy^{2^s}+ay^{2^s+1} \\ x^{2^{2s}+1}+ax^{2^{2s}}y+(1+a)^{2^s}xy^{2^{2s}}+ay^{2^{2s}+1} \end{pmatrix}$	**Field:** $n=2k$. **Exponent:** $\gcd(s,k)=1$. **Others:** $a \in \mathbb{F}_{2^k}$ s.t. $X^{2^s+1}+X+a$ has no root in $\mathbb{F}_{2^k}$.	[CLV22]
(CLV22b)	$(x,y) \mapsto \begin{pmatrix} x^3+xy+xy^2+ay^3 \\ x^5+xy+ax^2y^2+ax^4y+(1+a)^2xy^4+ay^5 \end{pmatrix}$	**Field:** $n=2k$. **Others:** $a \in \mathbb{F}_{2^k}$ s.t. X^3+X+a has no root in $\mathbb{F}_{2^k}$.	[CLV22]
(LK23a)	$(x,y,z) \mapsto \begin{pmatrix} x^{2^s+1}+x^{2^s}z+yz^{2^s} \\ x^{2^s}z+y^{2^s+1} \\ xy^{2^s}+y^{2^s}z+z^{2^s+1} \end{pmatrix}$	**Field:** $n=3k$. **Exponent:** $\gcd(s,k)=1$. **Others:** The polynomials of Theorem A.2 have no root in $\mathbb{F}_{2^k}$ or $\mathbb{F}_{2^k} \times \mathbb{F}_{2^k}$.	[LK23]
(LK23b)	$(x,y,z) \mapsto \begin{pmatrix} x^{2^s+1}+xy^{2^s}+yz^{2^s} \\ xy^{2^s}+z^{2^s+1} \\ x^{2^s}z+y^{2^s+1}+y^{2^s}z \end{pmatrix}$	**Field:** $n=3k$. **Exponent:** $\gcd(s,k)=1$. **Others:** The polynomials of Theorem A.3 have no root in $\mathbb{F}_{2^k}$ or $\mathbb{F}_{2^k} \times \mathbb{F}_{2^k}$.	[LK23]

Table A.4: Known infinite families of multivariate quadratic APN functions over $\mathbb{F}_{2^n}$ (2/2).

Appendix B

Reusing the tools of Chapter 6 for cryptanalysing an Sbox

So far, we only considered multiplicative decompositions of $\mathbb{F}_{2^{2k}}^*$. But as $\mathbb{F}_{2^k}$, with $n = 2k$ is an additive subgroup of $\mathbb{F}_{2^{2k}}$, we can also partition $\mathbb{F}_{2^{2k}}$ as:

$$\mathbb{F}_{2^{2k}} = \bigsqcup_{\lambda \in \mathcal{O}} \lambda + \mathbb{F}_{2^k},$$

given any system of representatives $\mathcal{O}$.

We refer to any $\lambda + \mathbb{F}_{2^k}$ as an *affine line* and to such λ as its *origin*. Any *set* $\mathcal{O}$ with 2^k elements such that $\mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}}(\mathcal{O}) = \mathbb{F}_{2^k}$ is a system of origins; and in particular the subspaces $\lambda\mathbb{F}_{2^k}$ with $\lambda \in \mathbb{F}_{2^{2k}} \setminus \mathbb{F}_{2^k}$.

B.1 Additive Subspace Property for Permutations

Let Γ be a system of multiplicative representative and γ° be the representative of $\mathbb{F}_{2^k}^*$. We denote by Γ° the set $\Gamma \setminus \{\gamma^\circ\}$. We now define a family of permutations Π, which map any multiplicative coset $\gamma\mathbb{F}_{2^k}^*$, $\gamma \neq \gamma^\circ$ onto the (punctured) additive coset $G(\gamma) + \mathbb{F}_{2^k}^*$, where $G(\gamma) \in \mathcal{O}$. Moreover, the restriction of Π to all cosets are the same, up to the additive offset, i.e., all the lines are shuffled the same way. Therefore, this property can be seen as "an additive variant" of the subspace property. However, if we want to construct a permutation of $\mathbb{F}_{2^{2k}}^*$, then for $\gamma = \gamma^\circ$, $\mathbb{F}_{2^k}^*$ must be mapped onto $\mathcal{O}$. As we will show later, the functions satisfying this property include, for $n = 8$, the Sbox used in the Russian standard primitives Streebog and Kuznyechik [Fed12, Fed15].

Definition B.1 (ASPP). A function $\Pi\colon \mathbb{F}_{2^{2k}} \to \mathbb{F}_{2^{2k}}$ with $\Pi(0) = 0$ is said to have the *additive subspace property for permutations (ASPP)* if there exist:

- a multiplicative system of representatives Γ and a system of origins $\mathcal{O}$, and
- two bijective maps: $G\colon \Gamma^\circ \to \mathcal{O}$, $F\colon \mathbb{F}_{2^k} \to \mathbb{F}_{2^k}$ with $F(0) = 0$, such that:

$$\Pi|_{\mathbb{F}_{2^{2k}} \setminus \mathbb{F}_{2^k}} : \gamma\varphi \mapsto G(\gamma) + F(\varphi) \text{ and } \Pi(\mathbb{F}_{2^k}^*) = \mathcal{O}. \tag{B.1}$$

Such a tuple $(\Gamma^\circ, \mathcal{O}, F, G)$ is called an *ASPP-decomposition of* Π. ▷

Since only Γ° matters in Definition B.1 and not Γ, γ° can always be chosen freely.

Proposition B.2. *Any function satisfying the ASPP is bijective.*

Proof. As it is defined, Π is injective on its restriction to $\mathbb{F}_{2^{2k}} \setminus \mathbb{F}_{2^k}$. By construction, F takes all values except 0 on $\mathbb{F}_{2^k}^*$. Thus, $\Pi(\mathbb{F}_{2^{2k}} \setminus \mathbb{F}_{2^k}) = \mathbb{F}_{2^{2k}}^* \setminus \mathcal{O}$. But the values in $\{0\} \cup \mathcal{O}$ are exactly the ones taken by Π on $\mathbb{F}_{2^{2k}}$. Π is thus a bijection. □

For any such functions, the decomposition is almost unique.

Definition B.3 (Trivially-equivalent decompositions)**.** Let Π be a function satisfying ASPP and $(\Gamma^\circ, \mathcal{O}, F, G)$ be a decomposition of Π. Let $\varphi \in \mathbb{F}_{2^k}^*$. Let $\widetilde{\Gamma^\circ} := \varphi\Gamma^\circ$, and $\widetilde{F} \colon \mathbb{F}_{2^k} \to \mathbb{F}_{2^k}$, $\widetilde{G} \colon \widetilde{\Gamma^\circ} \to \mathcal{O}$ be defined by:

$$\widetilde{F} = F \circ \mathrm{M}_\varphi \quad \text{and} \quad \widetilde{G} = G \circ \mathrm{M}_{\varphi^{-1}},$$

where the multiplication by φ (resp. by φ^{-1}) is denoted by M_φ (resp. $\mathrm{M}_{\varphi^{-1}}$). Then, the tuple $(\widetilde{\Gamma^\circ}, \mathcal{O}, \widetilde{F}, \widetilde{G})$ is an ASPP-decomposition of Π. $(\Gamma^\circ, \mathcal{O}, F, G)$ and $(\widetilde{\Gamma^\circ}, \mathcal{O}, \widetilde{F}, \widetilde{G})$ are called *trivially-equivalent.* ▷

The new tuple is a decomposition of Π. First, $\widetilde{F}, \widetilde{G}$ are well-defined and have the announced domains and codomains. Moreover, they are bijective as compositions of bijections, and $\widetilde{F}(0) = F(\varphi \times 0) = F(0) = 0$. By design, $\widetilde{\Gamma^\circ}$ is a system of directions without representative for $\mathbb{F}_{2^k}$. Finally, let $\widetilde{\gamma} \in \widetilde{\Gamma^\circ}, \psi \in \mathbb{F}_{2^k}^*$. Then $\widetilde{G}(\widetilde{\gamma}) + \widetilde{F}(\psi) = G(\varphi^{-1}\widetilde{\gamma}) + F(\varphi\psi)$. By construction $\widetilde{\gamma} = \varphi\gamma$ for some $\gamma \in \Gamma$. Thus:

$$\widetilde{G}(\widetilde{\gamma}) + \widetilde{F}(\psi) = G(\gamma) + F(\varphi\psi) = \Pi\left(\gamma\left(\varphi\psi\right)\right) = \Pi(\widetilde{\gamma}\psi).$$

Thus $(\widetilde{\Gamma^\circ}, \mathcal{O}, \widetilde{F}, \widetilde{G})$ satisfies Definition B.1 and is therefore a decomposition of Π. □

Proposition B.4 (Uniqueness of the decomposition)**.** *Let* Π *be a function satisfying the ASPP. Then all decompositions of* Π *are trivially equivalent.*

Proof. Let $(\Gamma^\circ, \mathcal{O}, F, G)$ and $(\widetilde{\Gamma^\circ}, \widetilde{\mathcal{O}}, \widetilde{F}, \widetilde{G})$ be two decompositions of Π. Because Γ° and $\widetilde{\Gamma^\circ}$ are systems of directions without representatives for $\mathbb{F}_{2^k}$, we can enumerate $\Gamma^\circ = \{\gamma_0, \cdots, \gamma_{2^k-1}\}$ and $\widetilde{\Gamma^\circ} = \left\{\widetilde{\gamma_0}, \cdots, \widetilde{\gamma_{2^k-1}}\right\}$ such that $\gamma_i = \widetilde{\gamma}_i\varphi_i$, where $\varphi_i \in \mathbb{F}_{2^k}^*$ for all $i \in [\![0, 2^k-1]\!]$.

Let $\psi \in \mathbb{F}_{2^k}^*$ and let $i \in [\![0, 2^k-1]\!]$. Then $\gamma_i\psi = \widetilde{\gamma}_i\varphi_i\psi$ and thus $G(\gamma_i) + F(\psi) = \widetilde{G}(\widetilde{\gamma}_i) + \widetilde{F}(\varphi_i\psi)$, so that:

$$F(\psi) + \widetilde{F}(\varphi_i\psi) = G(\gamma_i) + \widetilde{G}(\widetilde{\gamma}_i). \tag{B.2}$$

So, for any i, the restrictions of the functions F and $\widetilde{F} \circ \mathrm{M}_{\varphi_i}$ to $\mathbb{F}_{2^k}^*$ only differ by a constant, which is $F(1) + \widetilde{F}(\varphi_i)$. We can thus rewrite it as:

$$\forall \psi \in \mathbb{F}_{2^k}^*, \forall\, i \in \left[\!\left[0, 2^k-1\right]\!\right], \quad F(\psi) + F(1) = \widetilde{F} \circ \mathrm{M}_{\varphi_i}(\psi) + \widetilde{F}(\varphi_i). \tag{B.3}$$

The image of the restriction of the left-hand function to $\mathbb{F}_{2^k}^*$ is $\mathbb{F}_{2^k} \setminus \{F(1)\}$, and the image of the restriction of the right-hand function is $\mathbb{F}_{2^k} \setminus \left\{\widetilde{F}(\varphi_i)\right\}$. By

Eq. (B.3), those sets must coincide, so $F(1) = \widetilde{F}(\varphi_i)$ for all i. In particular, from the injectivity of $\widetilde{F}$, we get $\varphi_i = \varphi_j$ for all i, j. The single value taken by all φ_i is now denoted by $\varphi \in \mathbb{F}_{2^k}^*$. Consequently, we deduce that $\gamma_i = \widetilde{\gamma}_i\varphi$ for all i, and in particular:

$$\Gamma^\circ = \varphi\widetilde{\Gamma^\circ}.$$

Furthermore, by simplifying Eq. (B.3), we obtain that, for all $\psi \in \mathbb{F}_{2^k}^*$, $F(\psi) = \widetilde{F}(\psi\varphi)$.

Again, by simplifying Eq. (B.2), we observe that $G(\gamma_i) = \widetilde{G}(\widetilde{\gamma}_i)$ so that, for all i it holds that:

$$\widetilde{G} = G \circ \mathrm{M}_{\varphi^{-1}} \quad \text{but also} \quad \widetilde{\mathcal{O}} = \mathrm{Im}(\widetilde{G}) = \mathrm{Im}(G) = \mathcal{O}.$$

The two representations are therefore trivially equivalent. □

It is worth noting that, while the generalized cyclotomic property was independent of the choice of Γ, this is not the case for the ASPP.

B.2 Streebog/Kuznyechik Sbox

The Sbox used by the last two Russian standards, Streebog and Kuznyechik, is defined for $n = 8$, i.e., $\mathbb{F}_{2^{2k}} = \mathbb{F}_{256}$ and $\mathbb{F} = \mathbb{F}_{16}$. It is specified [Fed12, Fed15] as a look-up table of integers ranging from 0 to 255. In order to rather study a function $F\colon \mathbb{F}_{2^{2k}} \to \mathbb{F}_{2^{2k}}$, identification between $\mathbb{F}_{2^{2k}}$ and $[\![0, 255]\!]$ must be specified, as two different identifications gives two functions over $\mathbb{F}_{2^{2k}}$ with *a priori* different properties.

The latest study [Per19] on the Sbox points out a representation of $\mathbb{F}_{256}$ that we will also be using: $\mathbb{F}_{256} \simeq \mathbb{F}_2[X]/(X^8 + X^4 + X^3 + X^2 + 1)$. Identification between 8-bit words, integers of $[\![0, 255]\!]$ and polynomials of degree at most 7 is done canonically: $\sum_{i=0}^{7} b_i 2^i \simeq \sum_{i=0}^{7} b_i X^i \simeq (b_7, \cdots, b_0)$. Finally, we denote by $\Lambda\colon \mathbb{F}_{2^{2k}} \to [\![0, 255]\!]$ the isomorphism built from these relations. Through Λ, $\mathbb{F}_{2^{2k}}$ inherits from the ordering of integers.

In the following, we consider the bijective Sbox $\pi\colon \mathbb{F}_{2^{2k}} \to \mathbb{F}_{2^{2k}}$ defined by:

$$\pi = \Lambda^{-1} \circ \mathrm{LUT} \circ \Lambda,$$

where LUT is the look-up table given as specifications [Fed12, Fed15]. The previous studies [Per19, PU16, BPU16] of π show how much it interacts with both additive and multiplicative decompositions. We continue along this line of work by studying its normalized form $\pi_0 := \pi + \pi(0)$, which acts as a permutation of $\mathbb{F}_{2^{2k}}^*$.

The *TK-log* decomposition of π [Per19] can be restated by the fact that π_0 satisfies the ASPP. The decomposition of π is therefore unique and partially exhibited in [Per19]. Indeed, Perrin shows that the multiplicative coordinates correspond to the decomposition $\Gamma \times \mathbb{F}_{2^k}^*$ where $\Gamma = \{a^i, i \in [\![0, 16]\!]\}$ and a is a well-chosen root of the polynomial defining $\mathbb{F}_{2^{2k}}$, namely $a := \Lambda(2)$ as $2 \simeq X$. In other words, a is the class of X. Instead of Γ, we observe that $a^{17}\left\{a^{-i}, i \in [\![1, 16]\!]\right\} =$

Γ° and use the associated trivially-equivalent decomposition to describe some properties of π_0 that can be easily verified.

Proposition B.5. *Let $b = a^{-1}$. Let $(\Gamma^\circ, \mathcal{O}, F, G)$ be the ASPP-decomposition of π_0 with $\Gamma = \{b^i, i \in [\![0,16]\!]\} = \{1\} \cup \Gamma^\circ$. Let $G^\circ := \pi_0|_{\mathbb{F}_{2^k}}$.*

1. *Let $\lambda \in \mathbb{F}_{2^{2k}}$, and $\mathcal{O} \cap (\lambda + \mathbb{F}_{2^{2k}}) = \{o_\lambda\}$. Then $\forall \varphi \in \mathbb{F}_{2^k}$, $\Lambda(o_\lambda) \leq \Lambda(\lambda + \varphi)$, meaning that $\mathcal{O}$ is built by choosing the smallest possible element of each affine line as origin.*

2. *Let $\mathbb{F}_{2^k}$ be enumerated in increasing order as $\mathbb{F}_{2^k} = \{\varphi_0, \varphi_1, \cdots, \varphi_{15}\}$. Let $i \in [\![0,16]\!], j \in [\![0,14]\!]$. Then:*

$$\mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}} \circ \pi_0(b^{i+17j}) = \begin{cases} \mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}} \circ G(b^i) = \varphi_{i-1} & \text{if } i \neq 0 \\ \mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}} \circ G^\circ(b^{17j}) = \varphi_{j+1} & \text{if } i = 0 \end{cases}, \tag{B.4}$$

and $\mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}} \circ G^\circ(0) = 0 = \varphi_0$. In other words, enumerating both coordinates of preimages by increasing powers results in enumerating the origins of the images by increasing traces.

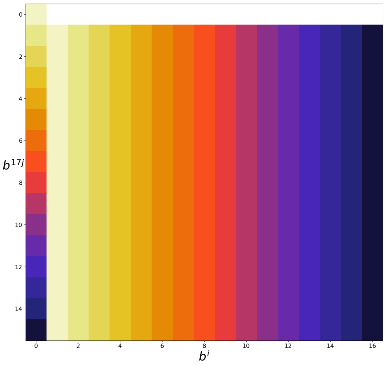


Figure B.1: Graphical representation of the values of $\mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}} \circ \pi_0(b^{i+17j})$. The first column corresponds to $i = 0$, i.e., to $\mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}} \circ \pi_0(\mathbb{F}_{2^k})$.

As pointed out in [Per19], $\mathcal{O}$ is even more structured as it is an $\mathbb{F}_2$-space of dimension 4. This structure is in line with Proposition B.5. Indeed, a natural way to obtain a system of representatives $\mathcal{O}$ is to complete any $\mathbb{F}_2$-basis of $\mathbb{F}_{2^k}$, $\mathcal{B}_0$, into a basis of $\mathbb{F}_{2^{2k}}$, $\mathcal{B}_0 \cup \mathcal{B}_1$. Then, $\mathcal{O} = \langle \mathcal{B}_1 \rangle$ is an additive system of representatives that is also an $\mathbb{F}_2$-subspace. The most natural algorithm to do so is to exhaust $\mathbb{F}_{2^{2k}} \setminus \mathbb{F}_{2^k}$ while keeping the first four vectors which make the rank grow. This procedure leads to the described system of origins. Regarding π_0, F is the least understood building-block. It remains an open question to determine whether a simple and natural description of F exists or not.

B.3 Walsh coefficients of the functions satisfying the ASPP

We now focus on the Walsh coefficients of the functions satisfying the ASPP. As for generalized cyclotomic mappings, we can express the Walsh coefficients of functions satisfying the ASPP in terms of the coefficients of their subfunctions.

Proposition B.6. *Let Π be a function satisfying the ASPP, $(\Gamma^\circ, \mathcal{O}, F, G)$ an ASPP-decomposition of Π, and $G^\circ = \Pi_{|\mathbb{F}_{2^k}}$. Let $\alpha \in \mathbb{F}_{2^{2k}}$ and $\beta \in \mathbb{F}_{2^{2k}}^*$ be decomposed as $\beta = \gamma_\beta \varphi_\beta$ with $\gamma_\beta \in \Gamma^\circ \cup \{1\}, \varphi_\beta \in \mathbb{F}_{2^k}^*$. Finally, let $G^\circ_{\gamma_\beta}$ be the function from $\mathbb{F}_{2^k}$ to $\mathbb{F}_{2^k}$ defined by $x \mapsto \mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}}(\gamma_\beta G^\circ(x))$. Then $W_{\mathbb{F}_{2^{2k}},\Pi}(\alpha,\beta) = S_{G^\circ} + S_F$ where:*

$$S_{G^\circ} = W_{\mathbb{F}_{2^k},G^\circ_{\gamma_\beta}}\left(\mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}}(\alpha), \varphi_\beta\right) - W_{\mathbb{F}_{2^k},G^\circ_{\gamma_\beta}}(0, \varphi_\beta)$$

$$\textit{and } S_F = \sum_{\gamma \in \Gamma^\circ} (-1)^{\mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}}(\beta G(\gamma))} W_{\mathbb{F}_{2^k},F}\left(\mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}}(\alpha\gamma), \mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}}(\beta)\right).$$

Proof. $W_{\mathbb{F}_{2^{2k}},\Pi}(\alpha,\beta)$ can be divided as $W_{\mathbb{F}_{2^{2k}},\Pi}(\alpha,\beta) = S_{\mathbb{F}_{2^k}} + S_{\mathbb{F}_{2^{2k}}\setminus\mathbb{F}_{2^k}}$ where:

$$S_{\mathbb{F}_{2^k}} = \sum_{\varphi \in \mathbb{F}_{2^k}} \chi_{\mathbb{F}_{2^{2k}}}(\alpha\varphi + \beta\Pi(\varphi)) \quad \text{and}$$
$$S_{\mathbb{F}_{2^{2k}}\setminus\mathbb{F}_{2^k}} = \sum_{\gamma \in \Gamma^\circ} \sum_{\varphi \in \mathbb{F}_{2^k}^*} \chi_{\mathbb{F}_{2^{2k}}}(\alpha\gamma\varphi + \beta\Pi(\gamma\varphi));$$

by dividing the sum over $\mathbb{F}_{2^{2k}} = \mathbb{F}_{2^k} \sqcup (\mathbb{F}_{2^{2k}} \setminus \mathbb{F}_{2^k})$ into two and using multiplicative coordinates in the second sum. Let us now prove the announced formulas for both halves. First, by using the trace linearity and transitivity, and by decomposing $\beta = \gamma_\beta\varphi_\beta$, we obtain:

$$\begin{aligned}
S_{\mathbb{F}_{2^k}} &= \sum_{\varphi \in \mathbb{F}_{2^k}} \chi_{\mathbb{F}_{2^{2k}}}(\alpha\varphi + \beta\Pi(\varphi)) \\
&= \sum_{\varphi \in \mathbb{F}_{2^k}} \chi_{\mathbb{F}_{2^k}}\left(\varphi \mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}}(\alpha) + \varphi_\beta \mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}}(\gamma_\beta\Pi(\varphi))\right) \\
&= W_{\mathbb{F}_{2^k},G^\circ_{\gamma_\beta}}\left(\mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}}(\alpha), \varphi_\beta\right).
\end{aligned}$$

Regarding the second sum, we observe that:

$$\begin{aligned}
S_{\mathbb{F}_{2^{2k}}\setminus\mathbb{F}_{2^k}} &= \sum_{\gamma \in \Gamma^\circ} \sum_{\varphi \in \mathbb{F}_{2^k}^*} \chi_{\mathbb{F}_{2^{2k}}}(\alpha\gamma\varphi + \beta F(\varphi) + \beta G(\gamma)) \\
&= \sum_{\gamma \in \Gamma^\circ} \chi_{\mathbb{F}_{2^{2k}}}(\beta G(\gamma)) \sum_{\varphi \in \mathbb{F}_{2^k}^*} \chi_{\mathbb{F}_{2^{2k}}}(\alpha\gamma\varphi + \beta F(\varphi)) \\
&= \sum_{\gamma \in \Gamma^\circ} \chi_{\mathbb{F}_{2^{2k}}}(\beta G(\gamma)) \sum_{\varphi \in \mathbb{F}_{2^k}} \chi_{\mathbb{F}_{2^{2k}}}(\alpha\gamma\varphi + \beta F(\varphi)) - \sum_{\gamma \in \Gamma^\circ} \chi_{\mathbb{F}_{2^{2k}}}(\beta G(\gamma)) \\
&= \sum_{\gamma \in \Gamma^\circ} \chi_{\mathbb{F}_{2^k}}(\beta G(\gamma))\, W_{\mathbb{F}_{2^k},F}\left(\mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}}(\alpha\gamma), \mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}}(\beta)\right) - \sum_{\gamma \in \Gamma^\circ} \chi_{\mathbb{F}_{2^k}}(\beta G(\gamma));
\end{aligned}$$

where we use the decomposition of Π, change the sum over $\mathbb{F}_{2^k}^*$ into a sum over $\mathbb{F}_{2^k}$, observe that $F(0) = 0$ and finally use the trace linearity and transitivity.

Finally, because $G(\Gamma^\circ) = \Pi(\mathbb{F}_{2^k})$, we get:

$$\sum_{\gamma \in \Gamma^\circ} \chi_{\mathbb{F}_{2^{2k}}}(\beta G(\gamma)) = \sum_{\varphi \in \mathbb{F}_{2^k}} \chi_{\mathbb{F}_{2^{2k}}}(\beta \Pi(\varphi)) = W_{\mathbb{F}_{2^k}, G^\circ_{\gamma_\beta}}(0, \varphi_\beta).$$

The decomposition $W_{\mathbb{F}_{2^{2k}},\Pi}(\alpha, \beta) = S_{G^\circ} + S_F$ is then deduced from $S_{G^\circ} = S_{\mathbb{F}_{2^k}} - W_{\mathbb{F}_{2^k}, G^\circ_{\gamma_\beta}}(0, \varphi_\beta)$ and $S_F = S_{\mathbb{F}_{2^{2k}} \setminus \mathbb{F}_{2^k}} + W_{\mathbb{F}_{2^k}, G^\circ_{\gamma_\beta}}(0, \varphi_\beta)$ □

Corollary B.7 (The case $\beta \in \mathbb{F}_{2^k}^*$). *Let Π be a function satisfying the ASPP, $(\Gamma^\circ, \mathcal{O}, F, G)$ be an ASPP-decomposition of Π, and $G^\circ = \Pi_{|\mathbb{F}_{2^k}}$. Let $\alpha \in \mathbb{F}_{2^{2k}}^*$, and α^{-1} be decomposed as $\alpha^{-1} = \varphi_{\alpha^{-1}}\gamma_{\alpha^{-1}}$, $\gamma_{\alpha^{-1}} \in \Gamma, \varphi_{\alpha^{-1}} \in \mathbb{F}_{2^k}^*$. Let $\beta \in \mathbb{F}_{2^k}^*$. Then:*

$$W_{\mathbb{F}_{2^{2k}},\Pi}(\alpha, \beta) = \begin{cases} 0 & , \text{ if } \alpha \in \mathbb{F}_{2^k}^* \\ W_{\mathbb{F}_{2^k}, G_1^\circ}\left(\mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}}(\alpha), \beta\right) + 2^k \varepsilon_{\alpha,\beta} & , \text{ if } \alpha \neq \mathbb{F}_{2^k} \end{cases},$$

where $G_1^\circ : x \mapsto \mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}}(G^\circ(x))$, and $\varepsilon_{\alpha,\beta} = (-1)^{\mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}}\left(\beta G(\gamma_{\alpha^{-1}})\right)}$.

Proof. Since $\beta \in \mathbb{F}_{2^k}^*$, we apply Prop. B.6 with $\gamma_\beta = 1$ and $\varphi_\beta = \beta$. By construction, $\Pi(\mathbb{F}_{2^k}) = \mathcal{O}$ so $G_1^\circ = \mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}} \circ \Pi|_{\mathbb{F}_{2^k}}$ is bijective. As $\beta \neq 0$, we get $W_{\mathbb{F}_{2^k}, G_1^\circ}(0, \beta) = 0$. We deduce that:

$$S_{G^\circ} = W_{\mathbb{F}_{2^k}, G_1^\circ}\left(\mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}}(\alpha), \beta\right) .$$

Moreover, because $\mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}}(\beta) = 0$, the sum S_F becomes:

$$S_F = \sum_{\gamma \in \Gamma^\circ} \chi_{\mathbb{F}_{2^{2k}}}(\beta G(\gamma))\, W_{\mathbb{F}_{2^k}, F}\left(\mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}}(\alpha\gamma), 0\right) \tag{B.5}$$

We now use that $W_{\mathbb{F}_{2^k}, F}(u, 0) = 2^k \cdot \mathbf{1}_0(u)$ and $\mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}}(\alpha\gamma) = 0$ if and only if $\gamma \in \alpha^{-1}\mathbb{F}_{2^k} = \gamma_{\alpha^{-1}}\mathbb{F}_{2^k}$. In other words, when $\gamma \in \Gamma^\circ$, $\mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}}(\alpha\gamma) = 0$ if and only if $\gamma = \gamma_{\alpha^{-1}}$. We distinguish two cases.

The case $\alpha \in \mathbb{F}_{2^k}^*$. In that case, $\alpha^{-1} \in \mathbb{F}_{2^k}^*$ and $\gamma_{\alpha^{-1}} = \gamma^\circ$. Thus, $\mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}}(\alpha\gamma)$ is never zero when γ goes over Γ° and $S_F = 0$. Finally,

$$W_{\mathbb{F}_{2^{2k}},\Pi}(\alpha, \beta) = S_{G^\circ} = W_{\mathbb{F}_{2^k}, G_1^\circ}\left(\mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}}(\alpha), \beta\right) = W_{\mathbb{F}_{2^k}, G_1^\circ}(0, \beta) = 0.$$

Indeed, G_1°, is a bijection, and $\beta \neq 0$.

The case $\alpha \in \mathbb{F}_{2^{2k}} \setminus \mathbb{F}_{2^k}$. In that case, only a single term of the sum in Eq. (B.5) remains: $S_F = 2^k \chi_{\mathbb{F}_{2^{2k}}}(\beta G(\gamma_{\alpha^{-1}}))$, so that:

$$W_{\mathbb{F}_{2^k}, G_1^\circ}(0, \beta) = W_{\mathbb{F}_{2^k}, G_1^\circ}\left(\mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}}(\alpha), \beta\right) + 2^k \chi_{\mathbb{F}_{2^{2k}}}(\beta G(\gamma_{\alpha^{-1}})) .$$

□

We now consider a very specific choice of multiplicative system of representatives. Indeed, because $2^{2k}-1=(2^k+1)(2^k-1)$, the group $\mathbb{G}$ of order 2^k+1 exists within $\mathbb{F}_{2^{2k}}^*$. If a is a primitive element, then $\mathbb{G}=\left\langle a^{2^k-1}\right\rangle$. Moreover, $\gcd(2^k-1,2^k+1)=\gcd(2^k-1,2)=1$ because 2^k-1 is odd, so according to the Chinese Remainder Theorem, any non-zero element can be uniquely decomposed as $a^{i(2^k-1)+j(2^k+1)}$. This implies that $\mathbb{G}$ is a multiplicative system of representatives. This choice, known as the *polar coordinate system* or *polar representation* (see for instance [Car21, page 191]), has especially been studied to design Boolean functions with notable properties [CF08, CM10, Lou+12, Zhe+13]. More generally, any *set* S with 2^k+1 elements is a multiplicative system of representatives if and only if it satisfies $\Theta(S)=\mathbb{G}$ where $\Theta\colon \mathbb{F}_{2^{2k}}\to\mathbb{F}_{2^{2k}},\quad x\mapsto x^{2^k-1}$. In [Göl15], Göloğlu introduces the trace-0/trace-1 representation which corresponds to $\Gamma=\{1\}\cup\mathrm{Tr}^{-1}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}}(\{1\})$. In the following proposition, using the polar representation, we construct functions satisfying the ASPP and which have a linearity which is at most twice the smallest known linearity for an Sbox over $\mathbb{F}_{2^{2k}}$.

Proposition B.8. *Let* Π *be a function satisfying the ASPP, with* $(\Gamma^\circ,\mathcal{O},F,G)$ *such that* $\Gamma^\circ=\mathbb{G}\setminus\{1\}$ *where* $\mathbb{G}\subset\mathbb{F}_{2^{2k}}^*$ *is the group of order* 2^k+1 *and* $F=\mathrm{Id}$*. Then* $\mathcal{L}(F)\leq 2^{k+2}$*.*

Proof. Let $\alpha,\beta\in\mathbb{F}_{2^{2k}}^*$. We observe that $\left|W_{\mathbb{F}_{2^{2k}},\Pi}(\alpha,\beta)\right|\leq|S_{G^\circ}|+|S_F|$. First of all,

$$\begin{aligned}|S_{G^\circ}| &= \left|W_{\mathbb{F}_{2^k},G^\circ_{\gamma_\beta}}\left(\mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}}(\alpha),\varphi_\beta\right)-W_{\mathbb{F}_{2^k},G^\circ_{\gamma_\beta}}(0,\varphi_\beta)\right|\\ &\leq \left|W_{\mathbb{F}_{2^k},G^\circ_{\gamma_\beta}}\left(\mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}}(\alpha),\varphi_\beta\right)\right|+\left|W_{\mathbb{F}_{2^k},G^\circ_{\gamma_\beta}}(0,\varphi_\beta)\right|\\ &\leq 2^{k+1}.\end{aligned}$$

Then,

$$|S_F|=\left|\sum_{\gamma\in\Gamma^\circ}\chi_{\mathbb{F}_{2^{2k}}}(\beta G(\gamma))\,W_{\mathbb{F}_{2^k},\mathrm{Id}}\left(\mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}}(\alpha\gamma),\mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}}(\beta)\right)\right|.$$

So we focus on bounding the remaining sum. We obviously know that $W_{\mathbb{F}_{2^k},\mathrm{Id}}(u,v)=2^k\cdot\delta_{u,v}$ for any $u,v\in\mathbb{F}_{2^k}$. In our case, $\mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}}(\alpha\gamma)=\mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}}(\beta)$ if and only if $\gamma\in\alpha^{-1}(\beta+\mathbb{F}_{2^k})$. Let us suppose that there exists $\varphi\in\mathbb{F}_{2^k}$ such that $\gamma=\alpha^{-1}(\beta+\varphi)$. In that case, because $\gamma\in\mathbb{G}$, we get $(\alpha^{-1}(\beta+\varphi))^{2^k+1}=\gamma^{2^k+1}=1$. This means that $(\beta+\varphi)^{2^k+1}=\alpha^{2^k+1}$. The left-hand side can be rewritten as:

$$(\beta+\varphi)^{2^k+1}=(\beta+\varphi)(\beta+\varphi)^{2^k}=(\beta+\varphi)(\beta^{2^k}+\varphi^{2^k})=\beta^{2^k+1}+(\beta+\beta^{2^k})\varphi+\varphi^2;$$

by successively using the linearity of $x\mapsto x^{2^k}$, and the fact that $\varphi^{2^k}=\varphi$ because $\varphi\in\mathbb{F}_{2^k}$. All in all, we obtain $\varphi^2+\mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}}(\beta)\varphi+\beta^{2^k+1}=\alpha^{2^k+1}$. This quadratic equation in φ has at most 2 solutions (in $\mathbb{F}_{2^k}$), so in any case we get,

$$\left|\sum_{\gamma \in \Gamma^{\circ}} \chi_{\mathbb{F}_{2^{2k}}}\left(\beta G(\gamma)\right) W_{\mathbb{F}_{2^k},\mathrm{Id}}\left(\mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}}(\alpha\gamma), \mathrm{Tr}_{\mathbb{F}_{2^{2k}}/\mathbb{F}_{2^k}}(\beta)\right)\right| \leq 2 \times 2^k.$$

Finally, we obtain:

$$\left|W_{\mathbb{F}_{2^{2k}},\Pi}(\alpha,\beta)\right| \leq |S_{G^{\circ}}| + \left|S_{\mathbb{F}_{2^k}}\right| \leq 2^{k+1} + 2 \times 2^k = 2^{k+2}.$$

□

In practice, we can even build such permutations with $\mathcal{L}(F) \leq 3 \times 2^k$: for $n = 8, 10, 12, 14$ and 16 bits, we obtained linearities equal respectively to $44, 80, 156, 300$ and 568. It is very counter-intuitive and quite surprising to obtain such low linearities by using, as subcomponent, the identity mapping. This points out that this subclass of functions, and more generally functions satisfying ASPP, should deserve more attention.

Appendix C

Contexte et résumé des travaux

C.1 Contexte général

Cryptographie. La cryptographie est la science assurant la sécurité de l'information, qu'elle soit en transit (échanges lors de la connexion à un site internet) ou bien au repos (données enregistrées sur un disque dur). L'opération dite de *chiffrement* consiste à transformer un message clair en une forme inintelligible, tandis que le *déchiffrement* est le processus inverse. Pour que le *chiffré* obtenu soit réellement inintelligible, il faut que la transformation dépende d'une information complètement inconnue d'une personne extérieure à la communication. Cette information inconnue, appelée communément *clé*, peut être la transformation tout entière mais il est d'usage de considérer la transformation comme complètement publique. La clé est alors un paramètre de la transformation. Cette clé est aussi nécessaire pour le déchiffrement. On parle de *chiffrement symétrique* lorsque la même clé est utilisée pour les deux procédés et de *chiffrement asymétrique* lorsque les clés prennent des formes différentes. Cette thèse porte sur la cryptographie symétrique.

Standardisation. Les algorithmes cryptographiques utilisés aujourd'hui pour la plupart des communications en ligne sont en réalité en nombre restreint : il s'agit de *standards*. En particulier, ceux émis par les États-Unis d'Amérique via le National Institute for Standards and Technology (NIST) sont souvent amenés à être ultérieurement adoptés par d'autres acteurs nationaux ou internationaux. Par exemple, l'Avanced Encryption Standard [DR02] (AES) est l'algorithme de chiffrement symétrique le plus largement utilisé dans le monde depuis sa standardisation aux États-Unis en 2002 [Aes]. À cause de l'émergence de nouvelles technologies, ces standards doivent néanmoins régulièrement être mis à jour par les autorités publiques. C'est la raison pour laquelle, le NIST a récemment choisi en 2023 [Dob+19], après un processus de plus de 7 ans, de promouvoir l'algorithme Ascon [Dob+21] comme futur standard de cryptographie dite “légère”.

Cryptographie légère. Cette branche de la cryptographie symétrique vise notamment à répondre aux besoins de sécurisation des objets connectés et de “l’Internet des objets” (IoT). En effet, les algorithmes actuellement standardisés nécessitent des ressources substantielles en terme de temps de calcul et d’énergie et ne peuvent pas être utilisés par un objet ayant une faible puissance de calcul [BP17]. Par exemple, une primitive implémentée sur des microcontrôleurs ne peut pas exiger un grand nombre de cycles de CPU car le temps pris par son évaluation deviendrait alors un goulet d’étranglement pour l’application de plus haut niveau l’utilisant. De manière similaire, le nombre de portes logiques disponibles pour le chiffrement sur une puce RFID est extrêmement restreint.

La cryptographie légère correspond à ces cas d’utilisation. De très nombreux algorithmes légers [Bor+12, Bei+16, Ban+15, Dob+21, Ava+23] ont été proposés au cours des deux dernières décennies dans des conférences, journaux de référence, mais aussi dans des compétitions internationales [Cae13, Nis17] comme celle organisée par le NIST. Ces algorithmes sont de natures très diverses [BP17], tant au niveau du type de primitive de base (permutation, chiffrement par bloc avec tweak) que des détails de conception de celle-ci (ARX, fonction de tour quadratique, structure de chiffrement à flot). Malheureusement, ils ont un point commun : ils ne peuvent jamais être prouvés complètement sûrs. En effet, il n’est possible de vraiment faire confiance à un algorithme cryptographique que s’il résiste à une analyse de sécurité intense et continue, comme c’est par exemple le cas avec l’AES.

Cryptanalyse. Face à ce constat, il est donc nécessaire de ne jamais stopper l’effort de cryptanalyse. Pour cela, des méthodes désormais très classiques existent. C’est le cas par exemple de la cryptanalyse différentielle [BS91b, BS91a] qui étudie la distribution de probabilité de la différence entre deux chiffrés correspondant à des clairs dont la différence est choisie, ou encore de la cryptanalyse linéaire [TG92, Mat94] qui estime la distance entre la fonction de chiffrement et l’ensemble des fonctions affines. Ces méthodes continuent de mettre en avant des failles de sécurité dans des algorithmes récents, alors même que ceux-ci ont été conçus pour résister à ce type d’analyse. Même si ces méthodes sont désormais relativement bien comprises [CV95, BN13, Bey21, BR22], beaucoup de problèmes restent ouverts, notamment au sujet de l’existence et de la construction de fonctions qui résistent de manière optimale à une analyse différentielle [Car15].

Avec l’avènement de la cryptographie légère, de nouveaux types d’attaques ont été inventés afin de tirer profit des choix de conceptions qui ont, par construction, été choisis pour réduire les coûts d’implémentation. En effet, s’il est vite tentant d’utiliser des constructions très structurées, des états de petite taille ou encore des composants non-linéaires émanant de fonctions quadratiques afin de garantir un coût réduit, ces choix de conception sont autant de potentiels leviers dont peut tirer avantage le cryptanalyste.

Mise en contexte. Cette thèse s'inscrit donc dans ce contexte. Dans un premier temps, deux généralisations des attaques différentielles, toutes deux très différentes, sont analysées et appliquées à des chiffrements symétriques légers. Ces approches reposent néanmoins sur une utilisation commune et continue de la représentation, désormais classique [Can16, Car21], des *fonctions booléennes*, c'est à dire des fonctions dont les entrées et sorties sont des bits, comme des polynômes multivariés dont les coefficients sont également des bits. Dans un deuxième temps, c'est précisément ces objets mathématiques que sont les fonctions booléennes qui sont considérés, et en particulier les fonctions dites APN [NK93] (Almost Perfect Non-linear) qui résistent de manière optimale à l'analyse différentielle. Enfin, après l'analyse, nous nous tournons vers la conception de primitives de cryptographie symétriques pour des usages autres que la cryptographie légère.

C.2 Analyse différentielle d'ordre supérieur et application à Ascon

Analyse différentielle d'ordre supérieure. La première généralisation de l'analyse différentielle considérée dans cette thèse est appelée analyse différentielle d'ordre supérieur [Knu95, Lai94], ou analyse intégrale [KW02], ou encore "par cubes" [DS09]. Cette méthode s'intéresse à l'analyse très précise de la forme normale algébrique (ANF en anglais) du chiffrement, c'est à dire à sa représentation polynomiale mentionnée plus haut. En particulier, lorsqu'il est possible de détecter la présence (ou l'absence) d'un monôme précisément identifié dans l'ANF du chiffrement, il est tout de suite possible de distinguer la primitive d'une bijection aléatoire. C'est là une première faiblesse : dans une situation idéale, il est préférable que la suite des chiffrés successifs ressemblent à une suite complètement aléatoire. Lorsqu'à cette détection s'ajoute la connaissance *exacte* de l'expression du coefficient ciblé, en fonction de la clé secrète, il est alors possible de monter une attaque par recouvrement de clé. Dans ce cas, la clé secrète peut être (en partie) récupérée par un adversaire extérieur à la communication et la confidentialité est alors compromise. Puisque la fonction de chiffrement est considérée publique, son expression polynomiale l'est aussi. Il est donc en pratique nécessaire que cette expression soit aussi dense que possible, au point de ne pas pouvoir être stockée sous forme développée et de ne pas pouvoir être facilement analysée.

Application à Ascon. Les attaques différentielles d'ordre supérieur sont donc particulièrement redoutables lorsque les composants utilisés pour construire le chiffrement sont tous de bas degré, typiquement, tous linéaires ou quadratiques. Dans ce cas, le chiffrement (ou une version simplifiée) n'atteint pas toujours le degré ou la "densité" attendue pour une bijection aléatoire. C'est précisément le cas pour Ascon. Ce chiffrement est un chiffrement *authentifié* [Rog02] qui assure à la fois la confidentialité de la communication, mais qui est aussi muni, par construction, d'un moyen de certifier que le message n'a pas été modifié pendant la communication, et qu'il provient bien de la personne attendue. Pour cela, Ascon utilise un mode de chiffrement dit en *éponge* [Ber+07] qui est décrit en Figure C.1. Dans ce mode, une bijection publique $p\colon \{0,1\}^{320} \to \{0,1\}^{320}$ de 320 bits est utilisée et successivement composée plusieurs fois avec elle même pour obtenir $p \circ p \ldots \circ p$. Mais cette brique de base p est très creuse et de degré seulement 2, alors qu'une bijection de 320 variables peut monter, jusqu'au degré 319. Ce bas degré est une contrainte de construction pour obtenir de très bonnes performances, mais aussi une faiblesse algébrique. Nous montrons en effet avec ce travail que lorsque qu'une paire (k, N) (à gauche en Figure C.1) est utilisée pour chiffrer de l'ordre de 2^{40} messages, alors Ascon ne permet pas d'assurer la confidentialité de tous les futurs messages chiffrés avec cette même paire. En effet, dans ce cas, l'état noté Σ_{AD} est fixé (car il ne dépend que de k, N et de IV qui est une constante connue) et peut être récupéré complètement grâce à une attaque différentielle d'ordre supérieure visant la bijection interne $p^{r_{\mathrm{in}}}$ où $r_{\mathrm{in}} = 6$. Contrairement à une attaque d'ordre supérieur classique, cette attaque est basée sur la récupération d'une information *incomplète* sur des coefficients, qui permet donc de retrouver de l'information sur Σ_{AD} seulement de manière conditionnelle. L'avantage de cette méthode est qu'elle ne nécessite pas le calcul trop coûteux de la forme polynomiale développée. Cette attaque nécessite beaucoup de prérequis, notamment une mauvaise utilisation de N qui doit normalement être changé à chaque chiffrement d'un nouveau message, ainsi qu'un grand nombre de paires clair/chiffré *connues de l'attaquant*. En revanche, sa complexité en temps est aussi de l'ordre de 2^{40} et sa complexité en mémoire suffisamment basse pour être mise en œuvre en pratique. Ce travail collaboratif avec Anne Canteaut & Léo Perrin est publié dans le journal IACR Transactions on Symmetric Cryptology, 2022(4) [BCP22].

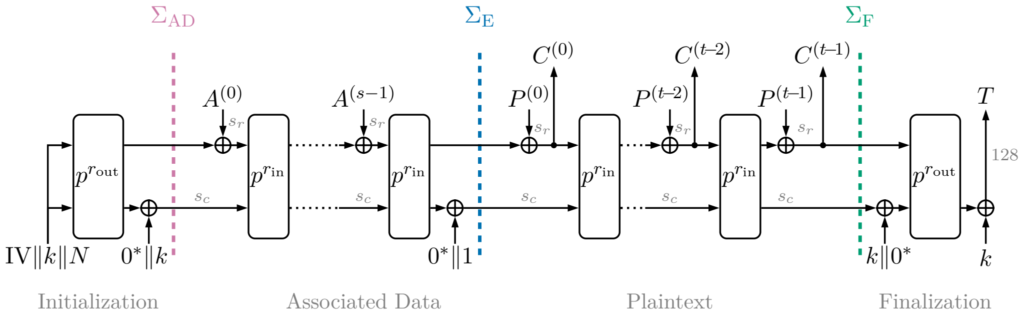


Figure C.1: Le mode de chiffrement authentifié d'Ascon.

C.3 Analyse commutative, analyse de chiffrements conjugués et analyse différentielle basée sur des lois de groupes alternatives

Attaques par invariants. La deuxième analyse considérée dans cette thèse s'inscrit quant à elle dans la lignée des attaques qui cherchent à mettre en avant une propriété restant invariante [Guo+16, TLS19, Bey18] à travers le chiffrement. En effet, la plupart des chiffrements ont une structure dite *itérée* où, comme dans Ascon mentionné plus haut, une fonction *de tour* est composée successivement avec elle-même. Lorsque qu'un adversaire peut établir qu'une propriété reste satisfaite avant ou après l'application de cette fonction, il n'est pas rare que cette propriété "traverse" le chiffrement tout entier. Par exemple, lorsqu'un ensemble de messages clairs $E \subset \{0,1\}^n$ vérifie $F(E) \subset E$ pour une fonction de tour $F\colon \{0,1\}^n \to \{0,1\}^n$ alors nécessairement $E \subset F^R(E)$ pour R itérations de F. Une telle propriété permet alors de nouveau de distinguer le chiffrement d'une bijection aléatoire puisqu'un message choisi dans E produira toujours un chiffré appartenant lui aussi à E. Le plus grand rempart à ce type de propriétés est l'ajout de clé de tour, c'est à dire le fait d'intercaler, entre deux applications de F, l'addition d'une *valeur dépendante* de la clé. En pratique, les clés de tours brisent ce type de symétries et réduisent à un petit nombre les clés pouvant satisfaire de telles invariances. Cela dépend néanmoins très largement de la complexité de l'algorithme de *cadencement* qui génère les clés de tours en fonction de la clé. Cette partie du chiffrement peut parfois être coûteuse et est souvent réduite au strict minimum dans les chiffrements légers. C'est par exemple le cas dans Midori [Ban+15], un chiffrement léger très inspiré de l'AES, où les clés de tours ne peuvent prendre que deux valeurs (qui dépendent de la clé) selon si l'indice du tour est pair ou impair. C'est une des raisons pour lesquelles les attaques par invariants sont si redoutables contre Midori [Bey18, BCL18, TLS19, Bey21]. Ce qui est beaucoup plus étonnant est que ce type d'attaque met aussi en évidence des faiblesses (pour certaines clés) qui relèvent du domaine de la cryptanalyse linéaire [Bey18], alors même que Midori est conçu pour favoriser la résistance à de telles attaques.

Analyse de Midori. Nous présentons dans ce travail une conclusion similaire vis à vis de faiblesses face à des attaques différentielles. Initialement, Midori peut être décrit sous la forme:

$$E_k := T_{k^{(R)}} \circ F \circ T_{k^{(R-1)}} \circ F \circ \ldots \circ F \circ T_{k^{(0)}},$$

où $k^{(i)}$ est la clé de tour i et $T_c\colon x \mapsto x + c$ est l'addition (ou translation) par c. Dans notre étude, nous considérons une représentation alternative, en utilisant *une bijection non-linéaire* G bien choisie servant de *changement de variables*. En effet, puisque $G^{-1} \circ G = \mathrm{Id}$, la fonction de chiffrement E_k pour la clé k peut être décrite comme:

$$E_k = T_{k^{(R)}} \circ G^{-1} \circ G \circ F \circ G^{-1} \circ G \circ T_{k^{(R-1)}} \ldots F \circ G^{-1} \circ G \circ T_{k^{(0)}}.$$

Un adversaire ayant le contrôle de l'entrée et de la sortie de E_k (sans pour autant connaitre k), peut dès lors tout à fait décider d'étudier $G \circ E_k \circ G^{-1}$, quitte à appliquer des transformations avant et après E_k. Autrement dit, un adversaire peut étudier un chiffrement *conjugué*, et, grâce aux applications $G \circ G^{-1}$ intercalées plus haut, une telle analyse peut être faite en étudiant les conjugués de la fonction de tour $G \circ F \circ G^{-1}$ et de l'addition de clé de tour $G \circ T_{k_i} \circ G^{-1}$. Une telle observation est d'ailleurs indépendante du type d'attaque effectuée et a déjà été appliquée [BCL18] à Midori dans le contexte de cryptanalyse linéaire. Dans le contexte différentiel, l'étude du conjugué de la fonction de tour, ou de ses composants (une fonction de tour étant souvent elle-même décomposable en sous-fonctions sous la forme $F = F_1 \circ F_2 \circ F_3$) est globalement similaire. La seule différence notable est qu'un composant linéaire, F_i, peut avoir un conjugué $G \circ F_i \circ G^{-1}$ non-linéaire puisque G et G^{-1}, sont par construction non-linéaires. En revanche, l'étude des conjugués de translations est elle très différente du cas usuel. En temps normal, étant donnée une paire $(x, x + d)$ qui diffère[1] de d, une telle paire devient $(x + c, x + c + d)$ après une translation par c, mais diffère toujours de d, quel que soit x. Cette propriété n'est *a priori* plus vérifiée par tous les conjugués $G \circ T_c \circ G^{-1}$, mais seulement pour certaines valeurs de c. Il s'agit donc d'une analyse qui s'applique uniquement à certaines clés *faibles*. Cette analyse fournit un distingueur déterministe qui permet de distinguer une version modifiée de Midori d'une bijection aléatoire avec seulement deux paires clair/chiffré *choisies*, lorsque Midori est utilisé avec une clé faible.

Multiples interprétations. Nous montrons aussi comment cette propriété peut s'interpréter comme la capacité des différents composants à *commuter* avec une application affine bien spécifique. Cette analyse peut de nouveau être faite composant par composant, mais cette fois-ci sur la description initiale du chiffrement. Cela permet notamment d'identifier ce phénomène comme un cas particulier d'auto-similarité [Bou+10, LMR15]. Une troisième interprétation est également possible, cette fois-ci comme une propriété différentielle pour une autre de loi de groupe que l'addition modulo 2, comme cela a déjà été proposé [CBS19], mais jamais réellement instancié sur un chiffrement grandeur nature. Enfin, dans le cadre différentiel classique, ce même distingueur met en avant des propriétés différentielles très étonnantes pour un chiffrement conçu pour y résister. La plupart de ces observations sont aussi applicables au chiffrement Scream. Une première partie de ce travail collaboratif avec Patrick Felke, Gregor Leander, Patrick Neumann, Léo Perrin & Lukas Stennes est publié dans le journal IACR Transactions on Symmetric Cryptology, 2022(4) [Bau+23]. Une deuxième partie, en collaboration avec les mêmes co-auteurs et Christof Beierle, est en cours de soumission.

[1] Notons que la différence modulo 2 est la même opération que l'addition modulo 2.

C.4 Analyse de fonctions vectorielles booléennes, propriétés des fonctions APN et cryptanalyse d'une boîte-S

Boites-S et fonctions APN. Dans un troisième temps, nous abordons l'analyse théorique de fonctions booléennes vectorielles utilisées en cryptographie et en particulier des fonctions non-linéaires de petite taille. Il s'agit là des fonctions avec peu de bits en entrée et en sortie qui sont communément appelées boîtes-S. Ces composants sont indispensables pour la conception de chiffrements par blocs et beaucoup d'arguments de sécurité, mais aussi de failles, sont très directement liés au choix des boîtes-S utilisées. Les fonctions dites APN [NK93] (Almost Perfect Non-linear) sont les fonctions qui résistent de manière optimale aux attaques différentielles. Ces objets optimaux sont très mal compris [Car15] et les exemples sont peu nombreux. En effet, à mesure que le nombre de bits en entrée/sortie augmente, les recherches exhaustives deviennent très vite impossible à mener et cette propriété n'est presque jamais vérifiée par une fonction tirée aléatoirement. Dès lors, l'analyse mathématique rigoureuse des fonctions APN reste l'une des seules options afin de mieux cerner cette propriété.

Fonctions APN et auto-équivalence. Dans cette optique, nous mettons en avant une propriété géométrique partagée par de nombreuses fonctions APN et en particulier par la plus mystérieuse de toutes, connue sous le nom de fonction de Kim [Bro+10]. Cette fonction APN de six variables (en entrée et en sortie) est en effet la seule à être équivalente (pour une notion d'équivalence bien précise) à une bijection APN, lorsque le nombre de variables est pair. Le "grand problème APN" posé il y a plus de 10 ans est de déterminer s'il existe ou non d'autres fonctions de ce type. Ce travail s'inscrit donc dans cet axe de recherche. La propriété dite de *cyclotomie*, partagée par de nombreuses fonctions APN, généralise la notion de fonction *monomiale*, c'est à dire une fonction pouvant être décrite par un polynôme univarié de la forme $P(X) = X^d$, dans un certain corps $\mathbb{F}_{2^n}$. Il est probable que cette propriété commune soit due au facteur humain. En effet, les premières fonctions APN connues étaient toutes monomiales [Nyb94, Kas71, Dob99a, Dob99b, Dob01] et les constructions qui ont suivies ont nécessairement été très largement inspirées des précédentes. Ce lien n'avait pourtant jamais été établi. Plus généralement, les fonctions cyclotomiques sont *auto-équivalentes*, une propriété partagée par la (quasi) totalité des fonctions APN connues. Nous tentons donc de clarifier ce lien entre le caractère APN et l'auto-équivalence. Une première partie de ce travail collaboratif avec Anne Canteaut & Léo Perrin a été présentée à la conférence internationale The Thirteenth International Workshop on Coding and Cryptography (WCC 2024) [BCP24]. Un article associé est en cours de soumission.

Cryptanalyse d'une boîte-S. Enfin, les outils de géométrie des corps finis de caractéristique 2 utilisés dans cette analyse permettent également l'analyse ad-hoc de certains composants de standards cryptographiques, lorsque les analyses des concepteurs sont lacunaires. Nous montrons en particulier comment ils peuvent préciser un peu plus la nature de la boîte-S, déjà préalablement passée au crible [Per19, PU16, BPU16], qui est utilisée dans le chiffrement par bloc russe Kuznyechik [Fed15] et la fonction de hachage cryptographique Streebog [Fed12].

C.5 Conception de primitives de cryptographie symétrique pour des usages émergents

Jeu d'instruction AES. Enfin dans un dernier temps, nous nous penchons sur la conception de primitives de cryptographie symétrique pour des usages émergents. Le premier cas est proche du cas de la cryptographie légère, mais cherche à obtenir de très hautes performances par des moyens différents. En effet, l'utilisation très répandue de l'AES a poussé les fabricants de processeurs, et Intel [Gue08] en particulier, à étendre le jeu d'instructions de certaines architectures d'ordinateurs afin d'y intégrer la fonction de tour de l'AES. Alors que celle-ci est loin d'être linéaire, sa vitesse est désormais comparable à celles de quelques OU EXCLUSIF logiques. La fonction de tour de l'AES est donc devenue ces dernières années un composant à part entière de nombreux algorithmes ultra-rapides, comme en témoigne la présence d'AEGIS-128 [WP14] et Deoxys-BC [Jea+21] parmi les candidats de la compétition CAESAR [Cae13]. Pourtant, aucun code d'authentification de message (MAC en anglais) *dédié*[2] n'a été conçu en se basant sur le jeu d'instructions AES-NI.

MACs basé sur l'AES. C'est le problème que nous abordons donc en premier. Les constructions de MACs que nous proposons sont basées sur un type particulier de primitives cryptographiques, les *fonctions de hachage universelles*, desquelles des MACs peuvent être dérivés en utilisant par exemple le *mode d'opération* EWCDM [CS16]. Nous portons une attention toute particulière aux garanties de sécurité des fonctions de hachage universelles construites, mais aussi aux performances de celles-ci. Pour cela, nous coordonnons et automatisons l'utilisation de deux solutions souvent dissociées : d'une part la résolution de problèmes linéaires en nombres entiers à l'aide d'un solveur qui nous permet d'estimer la résistance à la cryptanalyse différentielle, et d'autre part, la mesure précise des performances des candidats. En effet, la résolution d'un fastidieux problème d'optimisation peut-être évitée si le candidat ne permet pas d'obtenir des performances satisfaisantes. À l'inverse, il est inutile de tester les performances d'un candidat qui peut être très rapidement écarté à cause de notables problèmes de sécurité. Nous obtenons ainsi le MAC sécurisé atteignant les meilleurs performances à ce jour sur PC. Ce travail collaboratif avec Augustin Bariant, Gaëtan Leurent, Clara Pernot, Léo Perrin

[2] Il existe en revanche des chiffrements authentifiés basés sur l'AES, comme Tiaoxin [Nik14], qui peuvent être utilisés uniquement pour leur fonctionnalité d'authentification et ainsi jouer le rôle de MAC.

& Thomas Peyrin est publié dans le journal IACR Transactions on Symmetric Cryptology, 2022(4) [Bar+24].

Chiffrement homomorphe. Enfin, nous abordons le cas du *transchiffrement* dans le cadre d'un chiffrement asymétrique complètement homomorphe (FHE), souvent appelé plus simplement *chiffrement homomorphe*. Le chiffrement homomorphe permet en théorie de réaliser n'importe quelle opération sur des données chiffrées, sans jamais déchiffrer celles-ci. Un utilisateur pourrait donc en théorie demander à un moteur de recherche le chemin le plus court entre son domicile et son lieu de travail, sans que jamais le moteur n'ait accès aux-dites positions en clair. C'est donc autant de données personnelles qui pourraient être théoriquement protégées. Pendant longtemps, les solutions homomorphes proposées n'étaient pas réalisables en pratique. Le problème majeur était l'accumulation d'un bruit tout au long des calculs qui rendait le déchiffrement impossible après trop d'opérations. La technique dite de "bootstrapping" [Gen09] a permis de pallier ce problème en rafraichissant les chiffrés pendant le calcul, afin de ramener ce bruit à un niveau acceptable et de pouvoir continuer.

Pourtant, en pratique, les solutions actuelles sont toujours lentes et coûteuses, en particulier en terme de données échangées. En effet, les chiffrés FHE ont des tailles qui dépassent très largement celles des messages clairs initiaux, ce qui n'est pas le cas de chiffrés symétriques qui conservent la même taille.

Transchiffrement. Puisqu'un chiffrement FHE peut appliquer n'importe quel algorithme sur des données chiffrées (en FHE) et/ou des données publiques et retourner un chiffré FHE du résultat, il est tout à fait possible d'évaluer un déchiffrement, à la donnée d'une clé et d'un chiffré. Il est donc possible de chiffrer tous les messages initiaux en utilisant un chiffrement symétrique, de transmettre les chiffrés symétriques publiquement, la clé symétrique chiffrée en FHE et de calculer homomorphiquement le déchiffrement symétrique. Cela permet ainsi d'obtenir le chiffré FHE des données initiales en limitant la bande passante, puisque le seul chiffré FHE transmis est celui de la clé. La contrepartie est un temps de calcul allongé puisque le déchiffrement symétrique en FHE est désormais nécessaire avant toute opération sur les données. Il est donc important de limiter le coût de cette opération.

Bootstrapping. L'implémentation de FHE à laquelle nous nous intéressons se nomme TFHE [Chi+20] où le "T" signifie "tore". Dans ce contexte, les mesures de coût et les contraintes du chiffrement symétrique sont très différentes des cas d'utilisation classiques. En particulier, TFHE utilise nativement l'arithmétique modulaire, car il est basé sur le problème d'apprentissage avec erreurs, plus communément appelé LWE [Reg05]. De plus, cette implémentation est munie d'une opération de bootstrapping dite *programmable* qui permet, en plus du rafraichissement d'un chiffré, d'évaluer n'importe quelle fonction (donnée par sa table de valeurs), et ce, sans surcout. Dans notre cas, cela semble donc tout à fait propice à l'évaluation "gratuite" d'une boîte-S qui sera choisie seulement pour ses garanties de sécurité et non pour sa facilité d'implémentation.

Chiffrement à flot pour FHE. Dans ce contexte, nous présentons un chiffrement à flot basé sur le corps à 17 éléments et dont la construction s'inspire à la fois des chiffrements à flot classiques, mais aussi des chiffrements par blocs. Notre schéma utilise en effet une fonction de tour proche de celle de l'AES pour mettre à jour un état interne. Après chaque mise à jour, une clé de tour (générée par un registre à décalage à rétroaction linéaire, ou LFSR) est additionnée à l'état. Par la suite, certains mots de l'état sont extraits et masqués par l'addition de mots générés par un deuxième LFSR. La suite ainsi obtenue est la suite de masques de notre chiffrement. Grâce à une nouvelle approche inspirée des chiffrements par blocs, nous pouvons fournir des garanties de sécurités robustes, souvent difficiles à obtenir dans le cas des chiffrements à flots. Les performances du chiffrement sont de plus compétitives au vu des mesures de performance effectuées. Ce travail collaboratif avec Christina Boura, Nicolas Bon, Sonia Belaïd, Anne Canteaut, Gaëtan Leurent, Pascal Paillier, Léo Perrin, Matthieu Rivain & Yann Rotella est en cours de soumission à une conférence internationale.