Cryptography and Communications (2020) 12:821–840
https://doi.org/10.1007/s12095-020-00427-1

On the EA-classes of known APN functions in small dimensions

Marco Calderini[1]

Received: 4 September 2019 / Accepted: 27 February 2020 / Published online: 6 April 2020

Abstract
Recently Budaghyan et al. (Cryptogr. Commun. **12**, 85–100, 2020) introduced a procedure for investigating if CCZ-equivalence can be more general than EA-equivalence together with inverse transformation (when applicable). In this paper, we show that it is possible to use this procedure for classifying, up to EA-equivalence, all known APN functions in dimension 6. We also give some discussion for dimension 7, 8 and 9. In particular, in these cases it is possible to give an upper bound on the EA-classes contained in the CCZ-classes of the known APN functions.

Keywords EA-equivalence · CCZ-equivalence · Boolean functions · APN

Mathematics Subject Classification (2010) 94A60 · 06E30 · 14G50 · 11T71

1 Introduction

Symmetric cryptographic primitives and in particular block ciphers use substitution boxes (in brief, S-boxes) to bring "confusion" into the systems. Such confusion is necessary to prevent known attacks.

Given n and m two positive integers, the functions from $\mathbb{F}_{2^n}$ to $\mathbb{F}_{2^m}$ are called vectorial Boolean functions. Such functions are used as S-boxes in the design of block ciphers.

Among the properties that these functions have to satisfy we have a low differential uniformity (see definitions in Section 2) to allow resistance to the differential attack [2] and high nonlinearity to resist the linear attack [18]. The lowest differential uniformity for a vectorial Boolean function is 2. Functions reaching such lower bound are called Almost Perfect Nonlinear (APN).

The APN property (in general the differential uniformity) is preserved by different forms of equivalences between (vectorial) Boolean functions, such as EA-equivalence and

This article is part of the Topical Collection: *Boolean Functions and Their Applications IV*
Guest Editors: Lilya Budaghyan and Tor Helleseth

✉ Marco Calderini
marco.calderini@uib.no

[1] Department of Informatics, University of Bergen, PB 7803, Bergen, 5020, Norway

CCZ-equivalence. Since EA-equivalence is a particular case of CCZ-equivalence, it is possible to partition the space of all functions $\mathbb{F}_{2^n} \to \mathbb{F}_{2^m}$ into CCZ-equivalence classes and then partition each CCZ-equivalence class into EA-equivalence classes. For brevity, we will refer to these as "EA-class" and "CCZ-class". It was shown by Budaghyan et al. [3] that for quadratic APN functions CCZ-equivalence is more general than EA-equivalence together with taking inverses of permutations. In [7] the authors investigate further the relation between CCZ-equivalence and EA-equivalence with inverse transformation. While, in [9] the authors give a characterization of CCZ-equivalence in terms of twisting functions. Despite this, CCZ-equivalence is not yet fully well understood and, to the best of our knowledge, partitioning the CCZ-class of a function into its EA-classes is an hard task.

Classification of APN functions is, as well, a hard open problem. Complete classification for APN functions over $\mathbb{F}_{2^n}$ is known only for $n \leq 5$ [4], and for $n = 6$ the CCZ-classification of APN functions with algebraic degree at most 3 is known [15]. For $n \leq 5$, in [4], the authors give a classification of the APN functions up to EA-equivalence and CCZ-equivalence. For the case of $n = 6$, the classification of the known APN functions is given only up to CCZ-equivalence. The classification up to EA-equivalence is not known.

In this work, we use the procedure introduced in [7] for investigating the EA-classes contained in a CCZ-class of a given function. In order to do that, in Section 3 we give some propositions that can be used to improve the the procedure given in [7] and filter some of the results obtained from this procedure. We also obtain that the number of EA-classes contained in the CCZ-class of a function F is upper bounded by the number of simplex codes contained in a linear code associated to F.

In Section 4, we discuss relations between the different equivalence concepts for vectorial Boolean functions and code equivalence. We also introduce a new linear code that can be defined for the case of bijective maps that can be used to verify affine equivalence between two permutations, see Theorem 6.

For the case $n = 6$, in Section 5, we are able to give all the EA-classes of the known APN functions. We also studied further the case of the only APN permutation in even dimension [6]. For such a function we give the representatives of the EA-classes which contain a permutation and we also give the representatives of the affine classes (containing a permutation).

In Section 6, we extend our study also to dimension 7, 8 and 9 (for this last case we focus only on non-Gold APN power functions). In these dimensions checking EA-equivalence, which is based on some code equivalence, requires an amount of computing which is huge, but we are able to give an upper bound on the number of EA-classes. Moreover, for the case of non-Gold APN power functions we can determine the exact number of the EA-classes.

2 Preliminaries

Let $n \geq 2$, we denote by $\mathbb{F}_{2^n}$ the finite field with 2^n elements, by $\mathbb{F}_{2^n}^\star$ its multiplicative group and by $\mathbb{F}_{2^n}[x]$ the polynomial ring defined over $\mathbb{F}_{2^n}$. Any function $F : \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ can be represented as a univariate polynomial of degree at most $2^n - 1$ in $\mathbb{F}_{2^n}[x]$, that is

$$F(x) = \sum_{i=0}^{2^n-1} c_i x^i, \quad c_i \in \mathbb{F}_{2^n}.$$

For any i, $0 \leq i \leq 2^n - 1$, the *2-weight* of i is the (Hamming) weight of its binary representation. The algebraic degree of a function F is equal to the maximum 2-weight of the

exponent i such that $c_i \neq 0$. Functions of algebraic degree 1 are called *affine* and of degree 2 *quadratic*. Linear functions are affine functions without the constant term and they can be represented as $L(x) = \sum_{i=0}^{n-1} c_i x^{2^i}$. We denote the *trace* function by

$$Tr(x) = x + x^2 + \cdots + x^{2^{n-1}}.$$

Let $\lambda \in \mathbb{F}_{2^n}^\star$ and F be a function from $\mathbb{F}_{2^n}$ to itself, the λ-component of F is the Boolean function $F_\lambda : \mathbb{F}_{2^n} \to \mathbb{F}_2$ with $F_\lambda(x) = Tr(\lambda F(x))$.

For any function $F : \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ we denote the *Walsh transform* in $a, b \in \mathbb{F}_{2^n}$ by

$$\mathscr{W}_F(a, b) = \sum_{x \in \mathbb{F}_{2^n}} (-1)^{Tr(ax+bF(x))}.$$

For any Boolean function $f : \mathbb{F}_{2^n} \to \mathbb{F}_2$ the Walsh transform in $a \in \mathbb{F}_{2^n}$ is given by

$$\mathscr{W}_f(a) = \sum_{x \in \mathbb{F}_{2^n}} (-1)^{Tr(ax)+f(x)}.$$

The *Walsh spectrum* of a function F is the set of all possible values of the Walsh transform. The Walsh spectrum of a (vectorial) Boolean function F is strictly related to the notion of nonlinearity of F, denoted by $\mathscr{NL}(F)$, indeed we have

$$\mathscr{NL}(F) = 2^{n-1} - \frac{1}{2} \max_{a \in \mathbb{F}_{2^n}, b \in \mathbb{F}_{2^n}^\star} |\mathscr{W}_F(a, b)|.$$

If $\mathscr{W}_f(0) = 0$ then the Boolean function is called balanced. For any function $F : \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ it is well know that F is a bijection if and only if all its component functions are balanced.

The concept of differential uniformity of a function F is related to the number of solutions of the equation $F(x + a) + F(x) = b$ for $a \in \mathbb{F}_{2^n}^\star$ and $b \in \mathbb{F}_{2^n}$.

Definition 1 For a function F from $\mathbb{F}_{2^n}$ to itself, and any $a \in \mathbb{F}_{2^n}^\star$ and $b \in \mathbb{F}_{2^n}$, we denote by $\delta_F(a, b)$ the number of solutions of the equation $F(x + a) + F(x) = b$. The maximum value δ among the $\delta_F(a, b)$'s is called the differential uniformity of F, and F is said to be differentially δ-uniform. A function F is called almost perfect nonlinear (APN) if $\delta = 2$.

There are several equivalence relations of functions for which the differential uniformity (and thus the APN property) is preserved. Two functions F and F' from $\mathbb{F}_{2^n}$ to itself are called:

- *affine equivalent* if $F' = A_1 \circ F \circ A_2$ where the mappings $A_1, A_2 : \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ are affine permutations;
- *extended affine equivalent* (EA-equivalent) if $F' = F'' + A$, where the mappings $A : \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ is affine and F'' is affine equivalent to F;
- *Carlet-Charpin-Zinoviev equivalent* (CCZ-equivalent) if for some affine permutation $\mathscr{L}$ of $\mathbb{F}_{2^n} \times \mathbb{F}_{2^n}$ the image of the graph of F is the graph of F', that is, $\mathscr{L}(G_F) = G_{F'}$, where $G_F = \{(x, F(x)) : x \in \mathbb{F}_{2^n}\}$ and $G_{F'} = \{(x, F'(x)) : x \in \mathbb{F}_{2^n}\}$.

Obviously, the affine equivalence is included in EA-equivalence, and it is also well known that EA-equivalence is a particular case of CCZ-equivalence and every permutation is CCZ-equivalent to its inverse [10].

Recently, Yoshiara [20] and Dempwolff [12] have shown, independently, that two power APN functions are CCZ-equivalent if and only if they are EA-equivalent or one is EA-equivalent to the inverse of the second one. Moreover, for the case of quadratic APN functions, CCZ-equivalence coincides with EA-equivalence [19].

3 Properties and remarks on the CCZ-equivalence

In this section we will recall the procedure given in [7] and give some remarks and properties regarding CCZ-equivalence that will be useful in the investigation of the EA-classes contained in a CCZ-class.

Since we are interested in the EA-classes, without loss of generality, we assume that the affine permutation in the definition of CCZ-equivalence is linear. Indeed, using affine permutations instead of linear one we simply obtain a shift by a constant in the input and output of the resulting function (see for instance [7]).

Lemma 1 (**Lemma 3.1 in** [7]) *Let $L_1,\ L_2 : (\mathbb{F}_{2^n})^2 \to \mathbb{F}_{2^n}$ be linear maps and $a,\ b \in \mathbb{F}_{2^n}$, such that $\mathscr{L}(x, y) = (L_1(x, y) + a, L_2(x, y) + b)$ is a permutation. Let F and F' be CCZ-equivalent functions such that $\mathscr{L}$ maps the graph of F to the graph of F'. Then the linear part $\mathscr{L}'$ of $\mathscr{L}$ maps the graph of F to the graph of $F''(x) = F'(x + a) + b$.*

A linear map $\mathscr{L}$ defined over $(\mathbb{F}_{2^n})^2$ can be described as a formal matrix

$$\mathscr{L} = \begin{bmatrix} A_1 & A_2 \\ A_3 & A_4 \end{bmatrix}$$

where A_i are linear maps over $\mathbb{F}_{2^n}$ for $1 \le i \le 4$, and

$$\mathscr{L}(x, y) = \begin{bmatrix} A_1 & A_2 \\ A_3 & A_4 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} = (A_1(x) + A_2(y), A_3(x) + A_4(y)).$$

In particular,

$$F_1(x) = L_1(x, F(x)) = A_1(x) + A_2 \circ F(x) \tag{1}$$

and

$$F_2(x) = L_2(x, F(x)) = A_3(x) + A_4 \circ F(x). \tag{2}$$

From the definition of CCZ-equivalence we have that a linear permutation $\mathscr{L}$ is *admissible* for producing a CCZ-equivalent function from F if and only if $F_1(x)$ is a permutation. In terms of Walsh coefficients we have the following observation.

Observation 1 (**Observation 3.2 in** [7]) *The function F_1 in (1) is a permutation if and only if all its component are balanced, that is*

$$\mathscr{W}_{F_1}(0, \lambda) = \sum_{x \in \mathbb{F}_{2^n}} (-1)^{\mathrm{Tr}(\lambda A_1(x) + \lambda A_2 \circ F(x))} = 0, \quad \text{for all } \lambda \in \mathbb{F}_{2^n}^{\star}.$$

Denoting by L^* the adjoint operator of a linear map L (i.e. $Tr(yL(x)) = Tr(xL^*(y))$ for all $x, y \in \mathbb{F}_{2^n}$), we have

$$\mathscr{W}_{F_1}(0, \lambda) = \sum_{x \in \mathbb{F}_{2^n}} (-1)^{\mathrm{Tr}(A_1^*(\lambda)x + A_2^*(\lambda)F(x))} = \mathscr{W}_F(A_1^*(\lambda), A_2^*(\lambda)) = \mathscr{W}_{F_{A_2^*(\lambda)}}(A_1^*(\lambda)) = 0. \tag{3}$$

In [7], the authors introduce a procedure that permits to investigate the relation between CCZ-equivalence and EA-equivalence together with the inverse transformation (when applicable). Using this procedure it is possible, at least in small dimensions, to investigate the EA-classes contained in the CCZ-class of a given function.

The procedure given in [7] is useful for constructing linear permutations

$$\mathscr{L} = \begin{bmatrix} A_1 & A_2 \\ A_3 & A_4 \end{bmatrix}$$

mapping the graph of F onto the graph of another function F'. In particular, the procedure constructs the linear functions A_1 and A_2 defined over $\mathbb{F}_{2^n}$ so that $F_1(x) = L_1(x, F(x)) = A_1(x) + A_2 \circ F(x)$ is a permutation. Indeed, if we are able to construct L_1 with such a property, then it is always possible to determine L_2 in order to have $\mathscr{L}$ a linear permutation.

We are focusing on the EA-classes that are contained in the CCZ-class of some given function F. In the following, we will show some properties that permit to determine whether from two admissible permutation $\mathscr{L}$ and $\mathscr{L}'$ we can obtain EA-equivalent functions.

Remark 1 (*Remark 2 in* [3]) For a function $F : \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$, if $\mathscr{L} = (L_1, L_2)$ and $\mathscr{L}' = (L_1, L_2')$ are permutations such that the function $L_1(x, F(x))$ is a permutation, then the functions defined by the graphs $\mathscr{L}(G_F)$ and $\mathscr{L}'(G_F)$ are EA-equivalent.

This means that for all possible L_1, for covering the EA-classes of a given function F, we need to construct a single L_2.

Remark 1 can be easily extended with the following proposition.

Proposition 1 *Let F be a function over $\mathbb{F}_{2^n}$ and let*

$$\mathscr{L} = \begin{bmatrix} A_1 & A_2 \\ A_3 & A_4 \end{bmatrix}, \quad \mathscr{L}' = \begin{bmatrix} A_1' & A_2' \\ A_3' & A_4' \end{bmatrix}$$

be two linear permutations over $(\mathbb{F}_{2^n})^2$ such that $F_1(x) = L_1(x, F(x)) = A_1(x) + A_2 \circ F(x)$ and $F_1'(x) = L_1'(x, F(x)) = A_1'(x) + A_2' \circ F(x)$ are permutations. If $L_1'(x, y) = L \circ L_1(x, y)$ for some linear permutation L, then the functions defined by the graphs $\mathscr{L}(G_F)$ and $\mathscr{L}'(G_F)$ are EA-equivalent.

Proof Let $L_2(x, y) = A_3(x) + A_4(y)$. Since $L_1'(x, y) = L \circ L_1(x, y)$ then also $\mathscr{L}'' = (L_1', L_2)$ is a linear permutation and from Remark 1 we have that the functions defined by the graphs $\mathscr{L}'(G_F)$ and $\mathscr{L}''(G_F)$ are EA-equivalent.

Now,

$$\mathscr{L}'' = \begin{bmatrix} L & 0 \\ 0 & I \end{bmatrix} \cdot \mathscr{L},$$

where I is the identity map, which implies that the functions defined by the graphs $\mathscr{L}(G_F)$ and $\mathscr{L}''(G_F)$ are affine equivalent. □

We will show, now the procedure introduced in [7]. From now on, we consider a fixed basis $\{\beta_1, ..., \beta_n\}$ of $\mathbb{F}_{2^n}$ as vector space over $\mathbb{F}_2$.

For any $\lambda \in \mathbb{F}_{2^n}$ we define the set

$$\mathcal{ZW}(\lambda) = \{a \in \mathbb{F}_{2^n} \ : \ \mathscr{W}_{F_\lambda}(a) = 0\}.$$

Then we can define the following set

$$S_F = \{\lambda \in \mathbb{F}_{2^n}^\star \ : \ \mathcal{ZW}(\lambda) \neq \emptyset\} \cup \{0\}. \tag{4}$$

Note that if $L_1(x, y)$ is such that $F_1(x) = L_1(x, F(x)) = A_1(x) + A_2 \circ F(x)$ is a permutation then $\mathrm{Im}(A_2^*) \subseteq S_F$. So, any subspace U in S_F could be a possible candidate for $\mathrm{Im}(A_2^*)$.

Along this section we will denote by $\mathrm{Span}(v_1, \ldots, v_m)$ the vector (sub)space over $\mathbb{F}_2$ generated by the elements $v_1, \ldots, v_m \in \mathbb{F}_{2^n}$.

Procedure 2 (**Procedure 4.4 in** [7]) *Let $U \subseteq S_F$ be a subspace of dimension k. Let $\{u_1, ..., u_k\}$ be a fixed basis of U. Let A_2 be such that $A_2^*(\beta_i) = u_i$ if $1 \leq i \leq k$ and $A_2^*(\beta_i) = 0$ if $k + 1 \leq i \leq n$.*

For any $u \in U \setminus \{0\}$ we consider the set $\mathcal{ZW}(u)$, as defined before. To construct A_1 we need to determine the images, with the adjoint operator A_1^*, of the vectors β_i's. In order to do that, we need to select any possible k-tuple $a_1 \in \mathcal{ZW}(u_1)$, ..., $a_k \in \mathcal{ZW}(u_k)$ such that

(P1) $\sum_{i=1}^k \lambda_i a_i \in \mathcal{ZW}(\sum_{i=1}^k \lambda_i u_i)$ *for any $\lambda_1, ..., \lambda_k \in \mathbb{F}_2$, not all zero.*

These $a_1, ..., a_k$ will be the images by A_1^* of $\beta_1, ..., \beta_k$, respectively.

After that, for any of these k-tuples, we need to determine all possible $(n - k)$-tuples of elements $a_{k+1}, ..., a_n$ satisfying:

(P2) *$a_{k+1}, ..., a_n$ are linearly independent;*
(P3) *for any $a \in \mathrm{Span}(a_{k+1}, ..., a_n)\backslash\{0\}$, $a + \sum_{i=1}^k \lambda_i a_i \in \mathcal{ZW}(\sum_{i=1}^k \lambda_i u_i)$, for any $\lambda_1, \ldots, \lambda_k \in \mathbb{F}_2$.*

Remark 2 Condition **(P3)** is equivalent to have

$$\mathrm{Span}(a_{k+1}, ..., a_n) \subseteq \bigcap_{\lambda_i \in \mathbb{F}_2} \sum_{i=1}^k \lambda_i a_i + \mathcal{ZW}\left(\sum_{i=1}^k \lambda_i u_i\right),$$

where $a + \mathcal{ZW}(u) = \{a + v : v \in \mathcal{ZW}(u)\}$.

In the following we will give some observations in order to see how it is possible from Procedure 2 to obtain the EA-classes contained in the CCZ-class of a given function.

Observation 3 (**Observation 4.2 in** [7]) *Let $\{u_1, \ldots, u_k\}$ be any fixed basis of U (where k is the dimension of U), we can suppose that $A_2^*(\beta_i) = u_i$ for $i = 1, ..., k$ and* $\ker(A_2^*) = \mathrm{Span}(\beta_{k+1}, ..., \beta_n)$.

Indeed, suppose A_2^* is such that $A_2^*(w_i) = u_i$ for $i = 1, ..., k$ and $\ker(A_2^*) = \mathrm{Span}(w_{k+1}, ..., w_n)$ *for some $w_1, ..., w_n$ linearly independent. Then, we can consider the linear permutation L such that $L^*(\beta_i) = w_i$ for all i. Now, if $F_1(x) = A_1(x) + A_2(F(x))$ is a permutation, we can consider $F_1' = L \circ F_1$, which is again a permutation, and ${A_2'}^* = (L \circ A_2)^*$ is s.t. ${A_2'}^*(\beta_i) = u_i$ for $i = 1, ..., k$ and* $\ker({A_2'}^*) = \mathrm{Span}(\beta_{k+1}, ..., \beta_n)$.

From the previous observation we have that if L_1 is such that $\mathrm{Im}(A_2^*) = U$, then from the procedure applied to the subspace U, with some fixed basis, we obtain at least one function L_1' such that $L_1 = L \circ L_1'$ for some linear permutation L. Thus, from Proposition 1 we obtain the same EA-class of L_1 from L_1'.

Observation 4 *From the procedure we can see that in (P3) we need to check the subspaces of dimension $n-k$ contained in $\bigcap_{\lambda_i\in\mathbb{F}_2}\sum_{i=1}^{k}\lambda_i a_i+\mathcal{ZW}\left(\sum_{i=1}^{k}\lambda_i u_i\right)$. If we have $W\subseteq\bigcap_{\lambda_i\in\mathbb{F}_2}\sum_{i=1}^{k}\lambda_i a_i+\mathcal{ZW}\left(\sum_{i=1}^{k}\lambda_i u_i\right)$, of dimension $n-k$, then we can consider only one basis of W for constructing the elements $a_{k+1}, ..., a_n$ in Procedure 2. Indeed, let $\{a_{k+1}, ..., a_n\}$ and $\{a'_{k+1}, ..., a'_n\}$ be two basis of W. Let A_1 and A'_1 constructed from the procedure applied to a fixed space U (and so also A_2 is fixed), such that $A_1^*(\beta_i)=A_1'^*(\beta_i)=a_i$ for $1\le i\le k$, and $A_1^*(\beta_j)=a_j$, $A_1'^*(\beta_j)=a'_j$ for $k+1\le j\le n$.*

Let $V=\mathrm{Span}(\beta_{k+1}, ..., \beta_n)$, the restriction of A_1^* and $A_1'^*$ over V, $A_1^*|_V$ and $A_1'^*|_V$, are bijections from V to W and thus $(A_1^*|_V)^{-1}$, $(A_1'^*|_V)^{-1}$ are well defined. Let L be a linear permutation such that $L^*(\beta_i)=\beta_i$ for $1\le i\le k$ and $L^*(\beta_j)=(A_1^*|_V)^{-1}(a'_j)$ for $k+1\le j\le n$ (note that $(A_1^*|_V)^{-1}(a'_j)\in V$ and they form a basis for V, so L is a permutation). Now it is easy to check that $A'_1(x)=L\circ A_1(x)$ and that $A_2(y)=L\circ A_2(y)$ implying that $A'_1(x)+A_2(y)=L(A_1(x)+A_2(y))$ and from Proposition 1 we will obtain the same EA-class from these functions.

From the same function A_2 we could obtain several L_1's. We will show how it is possible to filter some of the L_1 obtained from the procedure.

Proposition 2 *Let F be a function defined over $\mathbb{F}_{2^n}$ with no linear monomials. Let $\mathscr{L}=(L_1,L_2)$ and $\mathscr{L}'=(L'_1,L'_2)$ be two linear permutations over $(\mathbb{F}_{2^n})^2$ with $L_1(x,y)=A_1(x)+A_2(y)$ and $L'_1(x,y)=A'_1(x)+A_2(y)$. Suppose $F_1(x)=L_1(x,F(x))$ and $F'_1(x)=L'_1(x,F(x))$ are permutations and the linear codes $\mathscr{C}_{F_1}$ and $\mathscr{C}_{F'_1}$ are equal, where the code $\mathscr{C}_F$ is generated by the matrix having as columns the vectors*

$$\big(\,F(x)\,\big)_{x\in\mathbb{F}_{2^n}}.$$

Then, if $\mathrm{Span}(\mathrm{Im}(A_2\circ F))=\mathrm{Im}(A_2)$ the functions defined by the graphs $\mathscr{L}(G_F)$ and $\mathscr{L}'(G_F)$ are EA-equivalent.

Proof Since $\mathscr{C}_{F_1}=\mathscr{C}_{F'_1}$ then there exists a linear permutations over $\mathbb{F}_{2^n}$ such that $F'_1(x)=L\circ F_1(x)$. In particular, since F has no linear monomials then $L\circ A_2\circ F=A_2\circ F$ and $L\circ A_1=A'_1$. Moreover, we have that $\mathrm{Span}(\mathrm{Im}(A_2\circ F))=\mathrm{Im}(A_2)$. This means that there exist $x_1, ..., x_k$ such that $F(x_1), ..., F(x_k)$ are linearly independent and $A_2\circ F(x_1), ..., A_2\circ F(x_k)$ form a basis for $\mathrm{Im}(A_2)$. Then, $\mathrm{Span}(\{F(x_1), ..., F(x_k)\})\oplus\ker(A_2)=\mathbb{F}_{2^n}$ and thus $L\circ A_2(y)=A_2(y)$ for all $y\in\mathbb{F}_{2^n}$. From this, we can conclude that $L'_1=L\circ L_1$ and from Proposition 1 it follows that the functions defined by the graphs $\mathscr{L}(G_F)$ and $\mathscr{L}'(G_F)$ are EA-equivalent. □

For the case of functions F having nonlinearity different from zero we have that $\mathscr{C}_{F_1}=\mathscr{C}_{F'_1}$ is sufficient to guarantee EA-equivalence.

Proposition 3 *Let F be a function defined over $\mathbb{F}_{2^n}$ with $\mathscr{NL}(F)\neq 0$ ($F(0)=0$). Let $\mathscr{L}=(L_1,L_2)$ and $\mathscr{L}'=(L'_1,L'_2)$ be two linear permutations over $(\mathbb{F}_{2^n})^2$ with $L_1(x,y)=A_1(x)+A_2(y)$ and $L'_1(x,y)=A'_1(x)+A_2(y)$. Suppose $F_1(x)=L_1(x,F(x))$ and $F'_1(x)=L'_1(x,F(x))$ are permutations. If $\mathscr{C}_{F_1}=\mathscr{C}_{F'_1}$, where the code $\mathscr{C}_F$ is defined as in Proposition 2, then the functions defined by the graphs $\mathscr{L}(G_F)$ and $\mathscr{L}'(G_F)$ are EA-equivalent.*

Proof Consider the matrix of size $2n \times 2^n$ with columns the vectors

$$M = \begin{pmatrix} x \\ F(x) \end{pmatrix}_{x \in \mathbb{F}_{2^n}}.$$

Since $\mathscr{NL}(F) \neq 0$ then the rows of this matrix are linear independent. Now, since F_1 is a permutation, the rows of

$$\big(F_1(x) \big)_{x \in \mathbb{F}_{2^n}},$$

are linear independent and for any row there exists a unique way of combining the rows of M to get it. Thus, there exist a unique linear function $L_1(x, y)$ such that

$$\big(F_1(x) \big)_{x \in \mathbb{F}_{2^n}} = \big(L_1(x, F(x)) \big)_{x \in \mathbb{F}_{2^n}}.$$

Since $\mathscr{C}_{F_1} = \mathscr{C}_{F_1'}$ we have that there exists a linear permutation L such that

$$\big(L \circ F_1(x) \big)_{x \in \mathbb{F}_{2^n}} = \big(F_1'(x) \big)_{x \in \mathbb{F}_{2^n}},$$

and then

$$\big(L \circ L_1(x, F(x)) \big)_{x \in \mathbb{F}_{2^n}} = \big(L_1'(x, F(x)) \big)_{x \in \mathbb{F}_{2^n}}.$$

From the unicity of L_1 and L_1' we obtain that $L_1' = L \circ L_1$. □

Remark 3 For the case of APN functions we have that the $\mathscr{NL}(F) \neq 0$ and so we can use this last proposition for filtering the functions obtained from Procedure 2.

Recalling that a simplex code (defined over $\mathbb{F}_2$) is a linear code of length $2^n - 1$ dimension n and all non zero codewords of hamming weight 2^{n-1}, we have the following upper bound on the number of EA-classes contained in the CCZ-class of a function F.

Corollary 1 *Let F be a function defined over $\mathbb{F}_{2^n}$ with $\mathscr{NL}(F) \neq 0$ ($F(0) = 0$). Let $\mathscr{C}(F)$ be the code generated by*

$$\begin{pmatrix} x \\ F(x) \end{pmatrix}_{x \in \mathbb{F}_{2^n}^\star}.$$

Then, the number of EA-classes contained in the CCZ-class of F is upper bounded by the number of the simplex codes contained in $\mathscr{C}(F)$.

4 Equivalence relations and linear codes

The main cryptographic properties (e.g. the APN property, the nonlinearity, etc.) can be interpreted as conditions on some binary linear codes, as first shown in [10].

Let F be a vectorial Boolean function then we can define the following codes related to F.

– The code $\mathscr{C}_1(F)$ which is generated by

$$C_1(F) := \begin{pmatrix} 1 \\ x \\ F(x) \end{pmatrix}_{x \in \mathbb{F}_{2^n}},$$

the size of the matrix is $(2n + 1) \times 2^n$.

– The code $\mathscr{C}_2(F)$ which is generated by

$$C_2(F) := \begin{pmatrix} 1 & 0 \\ x & 0 \\ F(x) & y \end{pmatrix}_{x\in\mathbb{F}_{2^n}, y\in\mathbb{F}_{2^n}^\star}$$

the size of the matrix is $(2n+1)\times(2^{n+1}-1)$.

– The code $\mathscr{C}_3(F)$ which is generated by

$$C_3(F) := \begin{pmatrix} 1 & 0 & 0 \\ x & 0 & z \\ F(x) & y & 0 \end{pmatrix}_{x\in\mathbb{F}_{2^n},\, y,z\in\mathbb{F}_{2^n}^\star}$$

the size of the matrix is $(2n+1)\times(2^{n+1}+2^n-2)$.

The equivalence between two functions F and G can be expressed in terms of linear codes. In particular, the code $\mathscr{C}_1$ was introduced in [5] for determining the CCZ-equivalence between two functions. The codes $\mathscr{C}_2$ and $\mathscr{C}_3$ were introduced in [13] for the case of the EA- and affine-equivalence. We summarize the results of [5] and [13] in the following theorem.

Theorem 5 *Let F and G be two vectorial Boolean functions. Then we have:*

– *F is CCZ-equivalent to G iff $\mathscr{C}_1(F)$ and $\mathscr{C}_1(G)$ are equivalent (*[5, Theorem 6.2]*).*
– *F is EA-equivalent to G iff $\mathscr{C}_2(F)$ and $\mathscr{C}_2(G)$ are equivalent (*[13, Theorem 10]*).*
– *If F is not a permutation, F is affine-equivalent to G iff the codes $\mathscr{C}_3(F)$ and $\mathscr{C}_3(G)$ are equivalent. If F is a permutation, F is affine-equivalent to G or G^{-1} iff the codes $\mathscr{C}_3(F)$ and $\mathscr{C}_3(G)$ are equivalent (*[13, Theorem 11]*).*

From the previous theorem when F and G are permutations we cannot distinguish if they are affine equivalent to each other or one is equivalent to the inverse of the other.

A necessary and sufficient condition for affine equivalence between permutations is the following.

Theorem 6 *Let F and G be two permutations over $\mathbb{F}_{2^n}$, with $n \geq 3$. F is affine-equivalent to G if and only if the codes $\mathscr{C}_4(F)$ and $\mathscr{C}_4(G+b)$ are equivalent for some $b \in \mathbb{F}_{2^n}$, where $\mathscr{C}_4(F)$ is generated by*

$$C_4(F) := \begin{pmatrix} 1 & 0 & 1 \\ x & 0 & z \\ F(x) & y & 0 \end{pmatrix}_{x,z\in\mathbb{F}_{2^n},\, y\in\mathbb{F}_{2^n}^\star}$$

of size $(2n+1)\times(2^{n+1}+2^n-1)$.

Proof Suppose that F is affine equivalent to G. Then, $B(F(Ax+a))+b = G(x)$ for some A, B linear permutations and $a, b \in \mathbb{F}_{2^n}$. Suppose that $b = 0$, otherwise we can consider the function $G' = G + b$.

Considering $L_1 = A^{-1}$, $L_2 = B$ linear permutations and $a' = A^{-1}a$ we have

$$M \cdot C_4(F) = \begin{pmatrix} 1 & 0 & 0 \\ a' & L_1 & 0 \\ 0 & 0 & L_2 \end{pmatrix} \times \begin{pmatrix} 1 & 0 & 1 \\ x & 0 & z \\ F(x) & y & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1 \\ L_1(x)+a' & 0 & L_1(z)+a' \\ L_2(F(x)) & L_2(y) & 0 \end{pmatrix},$$

applying a permutation on the columns, the last matrix is $C_4(G)$.

Vice versa, suppose that the code $\mathscr{C}_4(F)$ is equivalent to $\mathscr{C}_4(G')$, for some $G' = G + b$. We can suppose that $G' = G$ otherwise we will obtain the affine equivalence to G' which is equivalent to G.

Then, there exists a matrix

$$M = \begin{pmatrix} c & \mathbf{d} & \mathbf{e} \\ a & L_1 & L_2 \\ b & L_3 & L_4 \end{pmatrix}$$

and a permutation matrix P such that $M \cdot C_4(F) = C_4(G) \cdot P$. Thus, permuting the columns of $M \cdot C_4(F)$ we would be able to obtain the matrix $C_4(G)$. Now,

$$M \cdot C_4(F) = \left(\begin{array}{c|c|c} c + \mathbf{d} \cdot x + \mathbf{e} \cdot F(x) & \mathbf{e} \cdot y & c + \mathbf{d} \cdot z \\ \hline L_1(x) + L_2(F(x)) + a & L_2(y) & L_1(z) + a \\ \hline \underbrace{L_3(x) + L_4(F(x)) + b}_{\text{left part}} & \underbrace{L_4(y)}_{\text{middle part}} & \underbrace{L_3(z) + b}_{\text{right part}} \end{array}\right) \begin{array}{l} \}\text{upper part} \\ \}\text{center part} \\ \}\text{bottom part} \end{array}.$$

In the following we will refer to the different nine parts of the matrix as the left upper (LU) part, left center (LC) part, left bottom (LB) part, middle upper (MU) part, middle center (MC) part, middle bottom (MB) part, right upper (RU) part, right center (RC) part and right bottom (RB) part.

Now, we want to understand how we can permute the columns of the matrix above so that we can obtain $C_4(G)$. From that, we will be able to determine the structure of the matrix M.

First of all, note that the first row of the matrix must have the same weight as the first row of $C_4(G)$, that is 2^{n+1}. Suppose $\mathbf{d}, \mathbf{e} \neq 0$. Then $c + \mathbf{d} \cdot z$ and $\mathbf{e} \cdot y$ have weight 2^{n-1}, so $c + \mathbf{d} \cdot x + \mathbf{e} \cdot F(x)$ needs to be of weight 2^n. Let $S = \{y : \mathbf{e} \cdot y = 0\}$, which is a subspace of dimension $n - 1$. A column relative to any $y \in S$ needs to have $L_2(y) = 0$, because for obtaining $C_4(G)$ we cannot move this column in the left or right part. Thus, $\text{rank}(L_2) \leq 2$.

Moreover, all the columns of the y's not in S need to be moved in the left or right part since the first row in the middle part has to be equal to zero. For any column relative to some $y \notin S$, we have that $L_2(y) = r$ for some fixed r. But, in the LC and RC part we should obtain two times all the nonzero elements of $\mathbb{F}_{2^n}$, which would be not possible.

Suppose $\mathbf{d} = 0$, $\mathbf{e} \neq 0$. As before, let $S = \{y : \mathbf{e} \cdot y = 0\}$. Thus $L_2(y) = 0$ if $y \in S$ and $L_2(y) = r$ for some fixed r if $y \notin S$. Again, the columns of the y's not in S need to be moved in the left or right part, and we cannot obtain all the nonzero elements of $\mathbb{F}_{2^n}$ repeated two times in the center part.

Suppose that $\mathbf{d} \neq 0$, $\mathbf{e} = 0$ then we obtain only 2^n 1's on the first row, which is not possible.

Then, $c = 1$ and $\mathbf{d} = \mathbf{e} = 0$ and

$$M \cdot C_4(F) = \begin{pmatrix} 1 & 0 & 1 \\ L_1(x) + L_2(F(x)) + a & L_2(y) & L_1(z) + a \\ L_3(x) + L_4(F(x)) + b & L_4(y) & L_3(z) + b \end{pmatrix}.$$

Now, we have that for obtaining $C_4(G)$ we cannot permute the columns related to the middle part, involving the variable y, with the columns of the other parts. Thus $L_2 \equiv 0$ and

$$M \cdot C_4(F) = \begin{pmatrix} 1 & 0 & 1 \\ L_1(x) + a & 0 & L_1(z) + a \\ L_3(x) + L_4(F(x)) + b & L_4(y) & L_3(z) + b \end{pmatrix}.$$

Moreover, since in the MB part we should have all the nonzero elements of $\mathbb{F}_{2^n}$, and in the LC and RC part all the elements of $\mathbb{F}_{2^n}$, we have that L_1 and L_4 need to be permutations.

We need now to prove that $L_3(z)+b$ is constantly equal to 0. First, note that if $L_3(z)+b = b \neq 0$ for all z, then in the RB part of $M \cdot C_4(F)$ we would have all columns equals to b. Since we want to permute the columns of $M \cdot C_4(F)$ in order to obtain $C_4(G)$ (which has all zero columns on the RB part) this means that all the columns of the left part of $M \cdot C_4(F)$ should be permuted with the columns of the right part, implying $L_3(x)+L_4(F(x))+b \equiv 0$, which is not possible. Similarly if $L_3(z) + b$ is a permutation.

Suppose, then, that $L_3(z)+b$ is not null (and not constantly equal to b) or a permutation, which implies $\ker(L_3) \neq \{0\}, \mathbb{F}_{2^n}$. Then, in order to obtain $C_4(G)$ we should permute at least all the columns of the right part that are nonzero in the RB part (that involving $L_3(z)+b$) with some columns of the left part of the matrix for which $L_3(x)+L_4(F(x))+b$ is zero.

Now, let $S = \{z : L_3(z) + b \neq 0\} = \mathbb{F}_{2^n} \setminus \{z : L_3(z) + b = 0\}$. Denoting by $A(x) = L_3(x) + b$, since $\ker(L_3) \neq \{0\}, \mathbb{F}_{2^n}$ we have that for a given non zero element c in $\mathrm{Im}(A)$ there exist at least two elements $z_1, z_2 \in S$ such that $A(z_1) = A(z_2) = c$. Since for obtaining $C_4(G)$ we should permute (with the left part) all the columns of the right part related to the elements of S, we would obtain in the LB part two columns with the same value. But since both F and G are permutations then we cannot have repeated column here. Then, $L_3(x) + b$ needs to be constantly equal to 0.

So,

$$M \cdot C_4(F) = \begin{pmatrix} 1 & 0 & 1 \\ L_1(x) + a & 0 & L_1(z) + a \\ L_4(F(x)) & L_4(y) & 0 \end{pmatrix},$$

and thus

$$\begin{pmatrix} 1 \\ L_1(x) + a \\ L_4(F(x)) \end{pmatrix} = \begin{pmatrix} 1 \\ x \\ G(x) \end{pmatrix} \cdot P,$$

for some permutation matrix P, that is, $L_4(F(L_1^{-1}(x) + L_1^{-1}(a)) = G(x)$. □

These theorems on the relation between the equivalences defined for Boolean functions and the related codes are quite useful. For instance, the computer algebra package MAGMA implements a function for checking code equivalence, hence for small values of n can be possible to distinguish the different types of equivalence. Note that for the case of the affine-equivalence in [1] it is given an algorithm for checking it. We do not compare the complexity of checking the affine equivalence with codes and the algorithm given in [1]. However, the implementation with the code equivalence is very easy in MAGMA.

5 EA-classes in dimension 6

In this section we give the analysis carried out for the known APN functions in dimension 6. We used Procedure 2 for obtaining the admissible linear functions L_1. Then, comparing the codes relative to $L_1(x, F(x))$ we used Proposition 3 for filtering the maps L_1. After that EA-equivalence was tested using the linear code $\mathscr{C}_2(F)$.

In dimension 6 there are 14 known APN functions (13 are quadratics) up to CCZ-equivalence and they are listed in Table 1. In Table 1 we give also the number of EA-classes contained in the CCZ-class of each function, together with the degrees of the functions in the EA-classes. All the representatives of the EA-classes can be found in Appendix 1 of [8].

Table 1 CCZ-inequivalent APN functions over $\mathbb{F}_{2^6} = \langle \zeta \rangle$

N.	function	# EA-classes	Degrees
1	x^3	3	{⋆2,3,4⋆}
2	$x^3 + \zeta^{11}x^6 + ux^9$	3	{⋆ 2, 3, 4 ⋆}
3	$\zeta x^5 + x^9 + \zeta^4 x^{17} + \zeta x^{18} + \zeta^4 x^{20} + \zeta x^{24} + \zeta^4 x^{34} + \zeta x^{40}$	19	{⋆ 2, 3^^15, 4^^3 ⋆}
4	$\zeta^7 x^3 + x^5 + \zeta^3 x^9 + \zeta^4 x^{10} + x^{17} + \zeta^6 x^{18}$	13	{⋆2, 3^^9, 4^^3 ⋆}
5	$x^3 + \zeta x^{24} + x^{10}$	13	{⋆2, 3^^5, 4^^7 ⋆}
6	$x^3 + \zeta^{17}(x^{17} + x^{18} + x^{20} + x^{24})$	91	{⋆2, 3^^66, 4^^24 ⋆}
7	$x^3 + \zeta^{11}x^5 + \zeta^{13}x^9 + x^{17} + \zeta^{11}x^{33} + x^{48}$	19	{⋆2, 3^^15, 4^^3 ⋆}
8	$\zeta^{25}x^5 + x^9 + \zeta^{38}x^{12} + \zeta^{25}x^{18} + \zeta^{25}x^{36}$	85	{⋆2, 3^^66, 4^^18 ⋆}
9	$\zeta^{40}x^5 + \zeta^{10}x^6 + \zeta^{62}x^{20} + \zeta^{35}x^{33} + \zeta^{15}x^{34} + \zeta^{29}x^{48}$	91	{⋆2, 3^^63, 4^^27 ⋆}
10	$\zeta^{34}x^6 + \zeta^{52}x^9 + \zeta^{48}x^{12} + \zeta^6 x^{20} + \zeta^9 x^{33} + \zeta^{23}x^{34} + \zeta^{25}x^{40}$	91	{⋆2, 3^^66, 4^^24 ⋆}
11	$x^9 + \zeta^4(x^{10} + x^{18}) + \zeta^9(x^{12} + x^{20} + x^{40})$	86	{⋆2, 3^^69, 4^^16 ⋆}
12	$\zeta^{52}x^3 + \zeta^{47}x^5 + \zeta x^6 + \zeta^9 x^9 + \zeta^{44}x^{12} + \zeta^{47}x^{33} + \zeta^{10}x^{34} + \zeta^{33}x^{40}$	92	{⋆2, 3^^69, 4^^22 ⋆}
13	$\zeta(x^6 + x^{10} + x^{24} + x^{33}) + x^9 + \zeta^4 x^{17}$	85	{⋆2, 3^^66, 4^^18 ⋆}
14	$x^3 + \zeta^{17}(x^{17} + x^{18} + x^{20} + x^{24}) + \zeta^{14}(\mathrm{Tr}(\zeta^{52}x^3 + \zeta^6 * x^5 + \zeta^{19}x^7 + \zeta^{28}x^11 + \zeta^2 x^{13}) + (\zeta^2 x)^9 + (\zeta^2 x)^{18} + (\zeta^2 x)^{36} + x^{21} + x^{42})$	25	{⋆ 3^^10, 4^^15 ⋆}

5.1 Classification results for Dillon's APN permutation

Further analysis was done for the case of the Kim function $x^3 + \zeta x^{24} + x^{10}$. Indeed, this function is equivalent to a permutation [6]. This is the only known example of APN function equivalent to a permutation in even dimension.

In this case we studied also the affine equivalence classes. The reason why we are interested in this classification is that some characteristics of the vectorial Boolean functions, interesting for designing block ciphers, such as to be a permutation, the boomerang uniformity [11], the threshold implementation [17], etc., are invariants with respect the affine equivalence but not with respect to EA- and CCZ-equivalence.

Thus, classification of (bijective) vectorial Boolean functions up to affine equivalence is an important task.

Using the code equivalence we can see that in the CCZ-class of the Dillon's APN permutation we have 13 EA-classes with two of them containing a permutation, while the number of affine classes containing a permutation is 4.

Let

$$\begin{aligned} F_1(x) = {} & \zeta^{57}x^{60} + \zeta^{56}x^{58} + \zeta^{43}x^{57} + \zeta^{31}x^{56} + \zeta^{29}x^{53} + \zeta^{27}x^{52} + \zeta^{28}x^{51} + \zeta^{35}x^{50} + \zeta^{54}x^{49} + \\ & \zeta^{51}x^{48} + \zeta x^{46} + \zeta^{54}x^{44} + \zeta^{50}x^{43} + \zeta^{50}x^{42} + \zeta^{32}x^{41} + \zeta^{49}x^{40} + \zeta^{36}x^{39} + \zeta^{14}x^{38} + \zeta^{16}x^{37} + \\ & \zeta^{15}x^{35} + \zeta^{43}x^{34} + \zeta^{23}x^{33} + \zeta^{7}x^{32} + \zeta^{7}x^{30} + \zeta^{57}x^{29} + \zeta^{11}x^{26} + \zeta^{49}x^{25} + \zeta^{36}x^{24} + \zeta^{42}x^{23} + \\ & \zeta^{40}x^{22} + \zeta^{34}x^{21} + \zeta^{9}x^{20} + \zeta^{28}x^{19} + \zeta^{4}x^{18} + \zeta^{50}x^{17} + \zeta^{58}x^{16} + \zeta x^{15} + \zeta^{48}x^{14} + \zeta^{33}x^{13} + \\ & \zeta^{31}x^{12} + \zeta^{43}x^{11} + \zeta^{14}x^{10} + \zeta^{5}x^{9} + \zeta^{45}x^{8} + \zeta^{60}x^{7} + \zeta^{31}x^{6} + \zeta^{42}x^{5} + \zeta^{10}x^{4} + \zeta^{10}x^{3} + \zeta^{48}x, \end{aligned}$$

$$\begin{aligned} F_2(x) = {} & \zeta^{3}x^{60} + \zeta^{33}x^{58} + \zeta^{18}x^{57} + \zeta^{8}x^{56} + \zeta^{38}x^{53} + \zeta^{28}x^{52} + \zeta^{5}x^{51} + \zeta^{37}x^{50} + \zeta^{9}x^{49} + \zeta^{45}x^{48} + \\ & \zeta^{10}x^{46} + \zeta^{54}x^{44} + \zeta^{25}x^{43} + \zeta^{50}x^{42} + \zeta^{55}x^{41} + \zeta^{30}x^{40} + \zeta^{45}x^{39} + \zeta^{41}x^{38} + \zeta^{14}x^{37} + \zeta^{49}x^{36} + \\ & \zeta^{31}x^{35} + x^{34} + \zeta^{46}x^{33} + \zeta^{20}x^{32} + \zeta^{47}x^{30} + \zeta^{32}x^{29} + \zeta^{57}x^{28} + \zeta^{47}x^{26} + \zeta^{44}x^{25} + \zeta^{17}x^{24} + \\ & \zeta^{19}x^{23} + \zeta^{61}x^{22} + \zeta^{31}x^{21} + \zeta^{31}x^{20} + \zeta^{48}x^{19} + \zeta^{58}x^{18} + \zeta^{21}x^{17} + x^{16} + \zeta^{39}x^{15} + \zeta^{44}x^{14} + \\ & \zeta^{35}x^{13} + \zeta^{21}x^{12} + \zeta^{15}x^{11} + \zeta^{54}x^{10} + \zeta^{62}x^{9} + \zeta^{42}x^{8} + \zeta^{62}x^{7} + \zeta^{14}x^{6} + \zeta^{3}x^{5} + \zeta^{29}x^{4} + \\ & \zeta^{34}x^{3} + \zeta^{5}x^{2} + \zeta^{46}x, \end{aligned}$$

and

$$\begin{aligned} F_3(x) = {} & \zeta^{61}x^{60} + \zeta^{60}x^{58} + \zeta^{49}x^{57} + \zeta^{24}x^{56} + \zeta^{21}x^{54} + \zeta^{16}x^{53} + \zeta^{36}x^{52} + \zeta^{35}x^{51} + \zeta^{17}x^{50} + \\ & \zeta^{28}x^{49} + \zeta^{14}x^{48} + \zeta^{62}x^{46} + \zeta^{9}x^{45} + \zeta^{21}x^{44} + \zeta^{29}x^{43} + \zeta^{22}x^{42} + \zeta^{35}x^{41} + \zeta^{41}x^{40} + \\ & \zeta^{51}x^{39} + \zeta^{46}x^{38} + \zeta^{37}x^{37} + \zeta^{7}x^{36} + \zeta^{32}x^{35} + \zeta^{45}x^{34} + \zeta^{16}x^{33} + \zeta^{55}x^{32} + \zeta^{11}x^{30} + \\ & \zeta^{8}x^{29} + \zeta^{29}x^{28} + \zeta^{6}x^{27} + \zeta^{58}x^{26} + \zeta^{28}x^{24} + \zeta^{15}x^{23} + \zeta^{44}x^{22} + \zeta^{35}x^{21} + \zeta^{32}x^{20} + \\ & \zeta^{53}x^{19} + \zeta^{42}x^{18} + \zeta^{50}x^{17} + x^{16} + \zeta^{12}x^{15} + \zeta^{27}x^{14} + \zeta^{30}x^{13} + \zeta^{7}x^{12} + \zeta^{52}x^{11} + \\ & \zeta^{43}x^{10} + \zeta^{7}x^{9} + \zeta^{17}x^{8} + \zeta^{5}x^{7} + \zeta^{17}x^{6} + \zeta^{43}x^{5} + \zeta^{13}x^{4} + \zeta^{57}x^{3} + \zeta^{35}x^{2} + \zeta^{49}x. \end{aligned}$$

Then, the CCZ-class can be represented by F_1, the EA-classes containing a permutation can be given by F_1 and F_1^{-1}, and the affine-classes (always with a permutation) are represented by F_1, F_1^{-1}, F_2 and F_3. Note that with the code equivalence of the code $\mathscr{C}_3(F)$ we would obtain only 3 functions since F_1 is not affine equivalent to its inverse, while using $\mathscr{C}_4(F)$ we can distinguish the two functions.

Remark 4 F_2 and F_3 are affine-equivalent to their inverses.

For all the APN permutations we have that the degree of their components are

```
{* 3^^7, 4^^56 *}
```

and the Walsh spectrum of the single components is given by the multi-set

```
{*
{* -16, -8^^22, 0^^12, 8^^26, 16^^3 *}^^21,
{* -16^^2, -8^^20, 0^^12, 8^^28, 16^^2 *}^^21,
{* -16^^3, -8^^18, 0^^12, 8^^30, 16 *}^^7,
{* -16^^6, 0^^48, 16^^10 *}^^7,
{* -8^^24, 0^^12, 8^^24, 16^^4 *}^^7
*}
```

6 On the EA-classes of functions in dimension 7,8 and 9

For dimension 7 and 8 it is still possible to implement Procedure 2. Thus we can obtain at least one representative of each EA-class. However, checking EA-equivalence with the code equivalence require a huge amount of computations. Corollary 1 gives us an upper bound on the number of EA-classes based on the simplex codes contained in

$$\begin{pmatrix} x \\ F(x) \end{pmatrix}_{x\in\mathbb{F}_{2^n}}.$$

Using MAGMA we are able to provide the upper bound for all the known functions in $n = 7, 8$. Note that in dimension 7 and 8 we have a huge list of APN functions from [21]. For space reason here we give the upper bound only for the functions listed in [14].

6.1 n=7

In dimension 7, in [14] the authors listed 19 APN functions in Table 2, in [21] the authors found 471 new functions more. For the computer results on all these APN functions see Appendix 2 in [8].

Remark 5 For the x^{13}, x^{57} and x^{63} we can derive the exact number of EA-classes. Indeed, the two simplex subcodes individuated for each ones are those generated by

$$\big(F(x)\big)_{x\in\mathbb{F}_{2^n}} \quad \text{or} \quad \big(x\big)_{x\in\mathbb{F}_{2^n}}.$$

The representatives of the EA-classes that are related to these codes are F and F^{-1}. For x^{57} and x^{63} we have that they are cyclotomic equivalent (and thus affine equivalent) to their inverse, implying that the CCZ-class and and the EA-class coincide. For the case of x^{13}, its inverse is given by x^{88}. Since the cyclotomic classes of these two functions are distinct we can conclude that they are not EA-equivalent. Thus for x^{13} we have 2 EA-classes in the CCZ-class.

Table 2 CCZ-inequivalent APN functions over $\mathbb{F}_{2^7}$ given in [14]

N.	function	# EA-classes ≤
1	x^3	256
2	x^5	256
3	x^9	256
4	x^{13}	2
5	x^{57}	2
6	$x^{63}(inverse)$	2
7	$x^3 + \mathrm{Tr}(x^9)$	184
8	$x^{34} + x^{18} + x^5$	184
9	$x^{20} + x^6 + x^3$	324
10	$x^{66} + x^{34} + x^{20} + x^{17} + x^3$	184
11	$x^{34} + x^{33} + x^{17} + x^3$	184
12	$x^{34} + x^{33} + x^{10} + x^5 + x^3$	296
13	$x^{66} + x^{18} + x^9 + x^3$	212
14	$x^{33} + x^{17} + x^{12} + x^3$	240
15	$x^{66} + x^{34} + x^{20} + x^3$	184
16	$x^{72} + x^{40} + x^{12} + x^3$	184
17	$x^{72} + x^{40} + x^{34} + x^6 + x^3$	184
18	$x^{34} + x^{33} + x^{12} + x^6 + x^5 + x^3$	240
19	$x^{72} + x^{40} + x^{34} + x^6 + x^3 + \zeta^{27}(\mathrm{Tr}(\zeta^{20}x^3 + \zeta^{94}x^5 + \zeta^{66}x^9))$	216

6.2 n=8

In dimension 8 we have 23 functions in the table given in [14], see Table 3 (in [21] the authors found 8157 new functions more). We extend the computation also to the case of the inverse function that is 4-differentially uniform in this case.

Remark 6 For x^{57} we have only one simplex code, which implies that there is only one EA-class. As in dimension 7 for the inverse function x^{127} we have two simplex codes and these are generated by

$$\big(F(x)\big)_{x\in\mathbb{F}_{2^n}} \quad \text{or} \quad \big(x\big)_{x\in\mathbb{F}_{2^n}}.$$

These codes are relative to the class of F and of F^{-1}, thus we can conclude as before that the CCZ-class contains only one EA-class.

6.3 n=9

For this dimension we consider only the non-Gold APN power functions. We give the upper bound on the number of EA-classes in Table 4.

Remark 7 As before for x^{13}, x^{19} and x^{241} we have two simplex codes and two EA-classes for each function. For the inverse function x^{255} we have two simplex codes but only one EA-class.

Table 3 CCZ-inequivalent APN functions over $\mathbb{F}_{2^8}$ given in [14] and the inverse function

N.	function	# EA-classes $\leq$
1	x^3	256
2	x^9	256
3	x^{57}	1
4	$\zeta^{15}x^{48}+\zeta^{16}x^{33}+\zeta^{16}x^{18}+x^{17}+x^3$	256
5	$x^3+Tr(x^9)$	256
6	$x^9+Tr(x^3)$	256
7	$\zeta^{21}x^{144}+\zeta^{183}x^{66}+\zeta^{245}x^{33}+x^3$	256
8	$\zeta^{135}x^{144}+\zeta^{120}x^{66}+\zeta^{65}x^{18}+x^3$	256
9	$\zeta^{67}x^{192}+\zeta^{182}x^{132}+\zeta^{24}x^6+x^3$	256
10	$x^{160}+x^{132}+x^{80}+x^{68}+x^6+x^3$	464
11	$x^{66}+x^{40}+x^{18}+x^5+x^3$	368
12	$x^{130}+x^{66}+x^{40}+x^{12}+x^3$	400
13	$\zeta^{189}x^{192}+\zeta^{143}x^{144}+\zeta^{22}x^{132}+\zeta^{21}x^{129}+\zeta^{133}x^{96}+\zeta^{239}x^{72}+\zeta^{229}x^{66}+\zeta^{31}x^{48}+$ $\zeta^{187}x^{36}+\zeta^{185}x^{33}+\zeta^{68}x^{24}+\zeta^{236}x^{18}+\zeta^{75}x^{12}+\zeta^{91}x^9+\zeta^{97}x^6+\zeta^{160}x^3$ $\zeta^{100}x^{192}+\zeta^{12}x^{160}+\zeta^{15}x^{144}+\zeta^{243}x^{136}+\zeta^{234}x^{132}+\zeta^{33}x^{130}+\zeta^{39}x^{129}+\zeta^{139}x^{96}+$ $\zeta^{51}x^{80}+\zeta^{229}x^{72}+\zeta^{39}x^{68}+\zeta^{17}x^{66}+\zeta^{189}x^{65}+\zeta^{126}x^{48}+\zeta^{198}x^{40}+\zeta^{238}x^{36}+\zeta^{192}$	256
14	$x^{34}+\zeta^{217}x^{33}+\zeta^{122}x^{24}+\zeta^{144}x^{20}+\zeta^{169}x^{18}+\zeta^{141}x^{17}+\zeta^{236}x^{12}+$ $\zeta^{117}x^{10}+\zeta^{183}x^9+\zeta^{184}x^6+\zeta^{231}x^5+\zeta^{228}x^3$	400
15	$\zeta^{155}x^{192}+\zeta^{96}x^{144}+\zeta^{223}x^{132}+\zeta^{77}x^{129}+\zeta^{88}x^{96}+\zeta^{232}x^{72}+\zeta^{69}x^{66}+\zeta^{142}x^{48}+$ $\zeta^{168}x^{36}+x^{33}+\zeta^{145}x^{24}+\zeta^{234}x^{18}+\zeta^{202}x^{12}+\zeta^{94}x^9+\zeta^{189}x^6+\zeta^{241}x^3$	256

Table 3 (continued)

N.	function	# EA-classes ≤
16	$\zeta^{126}x^{192}+\zeta^{119}x^{144}+\zeta^{221}x^{132}+\zeta^{222}x^{129}+\zeta^{79}x^{96}+\zeta^{221}x^{72}+\zeta^{187}x^{66}+$ $\zeta^{148}x^{48}+\zeta^{187}x^{36}+\zeta^{237}x^{24}+\zeta^{231}x^{12}+\zeta^{119}x^{9}+\zeta^{244}x^{6}+\zeta^{236}x^{3}$	256
17	$\zeta^{151}x^{192}+\zeta^{13}x^{144}+\zeta^{58}x^{132}+\zeta^{143}x^{129}+\zeta^{110}x^{96}+\zeta x^{72}+\zeta^{244}x^{66}+\zeta^{26}x^{48}+$ $\zeta^{180}x^{36}+\zeta^{8}x^{33}+\zeta^{69}x^{24}+\zeta^{76}x^{18}+\zeta^{201}x^{12}+\zeta^{201}x^{9}+\zeta^{19}x^{6}+\zeta^{107}x^{3}$	256
18	$\zeta^{86}x^{192}+\zeta^{224}x^{129}+\zeta^{163}x^{96}+\zeta^{102}x^{66}+\zeta^{129}x^{48}+\zeta^{102}x^{36}+\zeta^{170}x^{33}+$ $\zeta^{14}x^{24}+\zeta^{170}x^{18}+\zeta^{101}x^{12}+\zeta^{58}x^{6}+\zeta^{254}x^{3}$	256
19	$\zeta^{95}x^{192}+\zeta^{242}x^{144}+\zeta^{195}x^{132}+\zeta^{98}x^{129}+\zeta^{84}x^{96}+\zeta^{45}x^{72}+\zeta^{234}x^{66}+\zeta^{202}x^{48}+$ $\zeta^{159}x^{36}+\zeta^{58}x^{33}+\zeta^{23}x^{24}+\zeta^{148}x^{18}+\zeta^{230}x^{12}+\zeta^{32}x^{9}+\zeta^{54}x^{6}+\zeta^{41}x^{3}$	256
20	$\zeta^{132}x^{192}+\zeta^{37}x^{144}+\zeta^{91}x^{132}+\zeta^{188}x^{129}+\zeta^{76}x^{96}+\zeta^{162}x^{72}+\zeta^{46}x^{66}+\zeta^{252}x^{48}+$ $\zeta^{42}x^{36}+\zeta^{81}x^{33}+\zeta^{83}x^{24}+\zeta^{13}x^{18}+\zeta^{185}x^{12}+\zeta^{163}x^{9}+\zeta^{216}x^{6}+\zeta^{181}x^{3}$	256
21	$\zeta^{91}x^{192}+\zeta^{124}x^{144}+\zeta^{214}x^{132}+\zeta^{106}x^{129}+\zeta^{59}x^{96}+\zeta^{172}x^{72}+\zeta^{138}x^{66}+$ $\zeta^{163}x^{48}+\zeta^{58}x^{36}+\zeta^{100}x^{33}+\zeta^{32}x^{24}+\zeta^{250}x^{18}+\zeta^{45}x^{12}+\zeta^{241}x^{6}+\zeta^{157}x^{3}$	256
22	$\zeta^{25}x^{192}+\zeta^{140}x^{144}+\zeta^{59}x^{132}+\zeta^{129}x^{129}+\zeta^{42}x^{96}+\zeta^{164}x^{72}+\zeta^{149}x^{66}+\zeta^{119}x^{48}+$ $\zeta^{74}x^{36}+\zeta^{211}x^{33}+\zeta^{9}x^{24}+\zeta^{46}x^{18}+\zeta^{130}x^{12}+\zeta^{185}x^{9}+\zeta^{147}x^{6}+\zeta^{27}x^{3}$	256
23	$\zeta^{113}x^{192}+\zeta^{56}x^{144}+\zeta^{68}x^{132}+\zeta^{155}x^{129}+\zeta^{91}x^{96}+\zeta^{78}x^{72}+\zeta^{159}x^{66}+\zeta^{30}x^{48}+$ $\zeta^{194}x^{36}+\zeta^{14}x^{33}+\zeta^{238}x^{24}+\zeta^{91}x^{18}+\zeta^{100}x^{12}+\zeta^{96}x^{9}+\zeta^{222}x^{6}+\zeta^{178}x^{3}$	256
-	x^{127} (inverse)	2

Table 4 CCZ-inequivalent APN functions over $\mathbb{F}_{2^9}$ given in [14] and the inverse function

N.	function	# EA-classes ≤
1	x^{13}	2
2	x^{19}	2
3	x^{241}	2
4	$x^{255}(inverse)$	2

In [16] the authors investigate EA-equivalence of the inverse function to a permutation. They concluded that for $n \geq 5$ the inverse function is EA-equivalent to a permutation if and only if it is affine equivalent to it. As the authors state at the end of their paper, an interesting problem is whether or not there exists a permutation that is CCZ-equivalent to x^{-1} but not affine equivalent. From our computational results we can conclude the following.

Theorem 7 *Let $5 \leq n \leq 9$. A permutation polynomial F defined over $\mathbb{F}_{2^n}$ is CCZ-equivalent to x^{-1} if and only if F is affine-equivalent to x^{-1}.*

Proof For $5 \leq n \leq 9$ we obtain only the two simplex codes generated by

$$\big(F(x)\big)_{x\in\mathbb{F}_{2^n}} \quad \text{or} \quad \big(x\big)_{x\in\mathbb{F}_{2^n}}.$$

This implies that we have only the EA-class of x^{-1} since it is an involution. Now, the permutations in the EA-class of x^{-1} can be obtained only with the affine equivalence [16]. □

From this result we give the following conjecture.

Conjecture 1 For $n \geq 5$, a permutation polynomial F defined over $\mathbb{F}_{2^n}$ is CCZ-equivalent to x^{-1} if and only if F is affine-equivalent to x^{-1}.

Moreover, in [7] the authors conjectured that the CCZ-class of non-Gold APN power functions can be obtained using iteratively EA-equivalence together with the inverse transformation. In particular, using Procedure 2 they proved that for $n \leq 8$ the conjecture is true. From the results obtained here we were able to verify that this is true up to dimension 9 and in particular we have at most two EA-classes whose representatives are F and F^{-1}.

Theorem 8 *Let $n \leq 9$ and $F(x) = x^d$ be a non-Gold APN function defined over $\mathbb{F}_{2^n}$. Then the CCZ-class of F is partitioned in at most two EA-classes represented by F and F^{-1} (when it exists).*

7 Conclusion

We gave the full classification, up to EA-equivalence, of the known APN functions in dimension 6 (see Table 1). Moreover, for the case of the unique APN permutation in even dimension, we gave also the classification of the affine classes (containing a permutation). For this purpose, in Theorem 6 we introduced a new code linked to a vectorial Boolean function that permits to investigate the affine equivalence in the context of bijective maps.

For dimension 7, 8 and 9, since checking EA-equivalence using the codes equivalence requires a huge amount of computing, we gave an upper bound on the number of the EA-classes of the known APN functions (in dimension 9 we consider only non-Gold APN power functions), see Tables 2, 3 and 4. For the case of APN power mapping we observed that at most we have two EA-classes in the CCZ-class, Theorem 8. Moreover, for the inverse function for $5 \leq n \leq 9$ we obtained that the EA-class coincides with the CCZ-class, implying that for these dimensions the inverse function is CCZ-equivalent to a permutation if and only if they are affine equivalent, Theorem 7.

Acknowledgements Open Access funding provided by University of Bergen. The research of this paper was supported by Trond Mohn Foundation.

References

1. Biryukov, A., Cannière, C.D., Braeken, A., Preneel, B.: A toolbox for cryptanalysis: linear and affine equivalence algorithms. In: EUROCRYPT, pp. 33–50 (2003)
2. Biham, E., Shamir, A.: Differential cryptanalysis of DES-like cryptosystems. J. Cryptol. **4**(1), 3–72 (1991)
3. Budaghyan, L., Carlet, C., Pott, A.: New classes of almost bent and almost perfect nonlinear polynomials. IEEE Trans. Inform. Theory **52.3**, 1141–1152 (2006)
4. Brinkmann, M., Leander, G.: On the classification of APN functions up to dimension five. Designs, Codes and Cryptography **49.1-3**, 273–288 (2008)
5. Browning, K.A., Dillon, J.F., Kibler, R.E., McQuistan, M.T.: APN polynomials and related codes. J. Combinatorics Inform. Sys. Sci. **34**(1-4), 135–159 (2009)
6. Browning, K.A., Dillon, J.F., McQuistan, M.T.: An APN permutation in dimension six. In: Contemporary Mathematics, vol. 518 (2010). (Post Proceedings of the Ninth International Conference on Finite Fields and Their Applications-Fq9), J. Am. Math. Soc., pp. 33–42
7. Budaghyan, L., Calderini, M., Villa, I.: On relations between CCZ- and EA-equivalences. Cryptogr. Commun. **12**, 85–100 (2020)
8. Calderini, M.: On the EA-classes of the known APN functions in small dimensions, ePrint Archive: Report 2019/369 (2019)
9. Canteaut, A., Perrin, L.: On CCZ-equivalence, extended-affine equivalence, and function twisting. Finite Fields Appl. **56**, 209–246 (2019)
10. Carlet, C., Charpin, P., Zinoviev, V.: Bent functions and permutations suitable for DES-like cryptosystems. Des. Codes Cryptogr. **15**, 125–156 (1998)
11. Cid, C., Huang, T., Peyrin, T., Sasaki, Y., Song, L.: Boomerang connectivity table: a new cryptanalysis tool. In: Nielsen, J.B., Rijmen, V. (eds.) EUROCRYPT 2018, Part II, vol. 10821 of LNCS, pp. 683–714. Springer, Heidelberg (2018)
12. Dempwolff, U.: CCZ equivalence of power functions. Designs, Codes and Cryptography **86**(3), 665–692 (2018)
13. Edel, Y., Pott, A.: On the equivalence of non-linear functions. Enhancing Cryptographic Primitives with Techniques from Error Correcting Codes, 87–103, NATO Sci. Peace Secur. Ser. D Inf. Commun. Secur., 23, IOS, Amsterdam (2009)
14. Edel, Y., Pott, A.: A new almost perfect nonlinear function which is not quadratic. Adv. Math. Comm. **3.1**, 59–81 (2009)

15. Langevin, P., Saygi, Z., Saygi, E.: Classification of APN cubics in dimension 6 over GF(2). http://langevin.univ-tln.fr/project/apn-6/apn-6.html (2020)
16. Li, Y., Wang, M.: Permutation polynomials EA-equivalent to the inverse function over $GF(2^n)$. Cryptogr. Commun. **3.3**, 175–186 (2011)
17. Nikova, S., Rechberger, C., Rijmen, V.: Threshold implementations against side-channel attacks and glitches. In: International Conference on Information and Communications Security, pp. 529–545. Springer, Berlin (2006)
18. Matsui, L.: Linear Cryptanalysis Method for DES Cipher. Advances in Cryptology EUROCRYPT93, pp. 386–397. Springer, Berlin (1994)
19. Yoshiara, S.: Equivalences of quadratic APN functions. Journal of Algebraic Combinatorics **35**, 461–475 (2012)
20. Yoshiara, S.: Equivalences of power APN functions with power or quadratic APN functions. Journal of Algebraic Combinatorics **44**(3), 561–585 (2016)
21. Yu, Y., Wang, M., Li, Y.: A matrix approach for constructing quadratic APN functions. Designs, Codes and Cryptography **73.2**, 587–600 (2014)