

New Instances of Quadratic APN Functions

Christof Beierle and Gregor Leander

Ruhr University Bochum, Bochum, Germany

Abstract

In a recent work, Beierle, Brinkmann and Leander presented a recursive tree search for finding APN permutations with linear self-equivalences in small dimensions. In this paper, we describe how this search can be adapted to find many new instances of quadratic APN functions. In particular, we found 12,921 new quadratic APN functions in dimension eight, 35 new quadratic APN functions in dimension nine and five new quadratic APN functions in dimension ten up to CCZ-equivalence. Remarkably, two of the 35 new APN functions in dimension nine are APN permutations.

Among the 8-bit APN functions, there are three extended Walsh spectra that do not correspond to any of the previously-known quadratic 8-bit APN functions and, surprisingly, there exist at least four CCZ-inequivalent 8-bit APN functions with linearity 2^7 , i.e., the highest possible non-trivial linearity for quadratic functions in dimension eight.

Keywords: almost perfect nonlinear, Walsh spectrum, linearity, self-equivalence, EA-equivalence

1 Introduction

Vectorial Boolean functions are used as S-boxes in many block ciphers and thus belong to the fundamental building blocks in symmetric cryptography. When such functions are used in actual cryptographic designs, one has to ensure that they fulfill certain criteria in order to prevent crypt-analytic attacks. Functions offering the best possible resistance against differential attacks [BS91] are called *almost perfect nonlinear (APN)*.

Definition 1. [NK92] *Let n be a positive integer. A function $F: \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ is called almost perfect nonlinear (APN) if, for every $a \in \mathbb{F}_2^n \setminus \{0\}, b \in \mathbb{F}_2^n$, the equation $F(x) + F(x + a) = b$ has at most 2 solutions $x \in \mathbb{F}_2^n$.*

This work was funded by Deutsche Forschungsgemeinschaft (DFG); project number 411879806 and by DFG under Germany's Excellence Strategy - EXC 2092 CASA - 390781972.

This is the version accepted to IEEE Transactions on Information Theory. DOI of the final published version: 10.1109/TIT.2021.3120698.

© 2021 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

We know several infinite families of APN functions as well as some sporadic instances. The vast majority of the known instances corresponds to monomial functions over the finite field \mathbb{F}_{2^n} or to functions with algebraic degree 2 over \mathbb{F}_2^n (i.e., quadratic functions). Indeed, at the time of writing, only a single APN instance is known that is not CCZ-equivalent to a monomial function or to a quadratic function [EP09]. The *linearity* is a measure for how well a function can be approximated by an affine function and is in particular of importance for resistance of block ciphers against linear cryptanalysis [Mat93]. Interestingly, APN functions often have low linearity as well. While this is always true for quadratic APN functions in odd dimension n (since those functions are almost bent [CCZ98]), the situation is more complicated for even values of n . For instance, prior to our work, two possible values of the linearity and three possible extended Walsh spectra of quadratic APN functions in dimension $n = 8$ were known [Ars18, Table 4.2]. Moreover, in dimension $n = 6$, we know one quadratic APN instance that admits the highest possible linearity of 2^{n-1} , see the list in [EP09]. It is an open question whether such a high linearity is also achievable by quadratic APN functions in higher dimensions [Car18].

A big open problem is to find new instances of APN *permutations* in even dimensions. Until now, only a single instance of an APN permutation in even dimension is known, namely for $n = 6$ [BDMW10]. It is well known that a quadratic APN function in even dimension cannot be a permutation [SZZ94]. However, the aforementioned permutation is CCZ-equivalent to a quadratic function. The potential to discover new APN permutations in even dimension that are CCZ-equivalent to quadratic functions is one motivation to explicitly search for quadratic APN functions.

For $n \leq 5$, a complete classification of APN functions up to CCZ-equivalence is known [BL08] and for $n = 6$, such a classification is known for APN functions up to algebraic degree three [Lan12]. In 2009, Edel and Pott introduced the switching construction and found new APN functions in dimensions $n \leq 8$ by replacing components of previously-known APN functions [EP09]. This led to the discovery of new APN functions, bringing up the number of known CCZ-inequivalent APN functions in dimension $n = 7$ and $n = 8$ to 19 and 23, respectively. A breakthrough was achieved by the works [WTG13] and [YWL14], in which the authors found many new quadratic APN instances in dimension $n = 7$ and $n = 8$. Recently, a new quadratic APN function in dimension 7 was found and a complete classification of quadratic APN functions in dimension 7 was achieved [KI20b, KI20a]. Excluding our results, there are 491 known APN instances in dimension $n = 7$ and 8,192 known APN instances in dimension $n = 8$ at the time of submission of this manuscript¹ in December 2020. Besides searching for APN instances in fixed dimensions, several infinite families of APN functions have been found, see [BCV20] for a recent summary.

In [BBL21], the authors utilized a recursive tree search for finding APN permutations with linear self-equivalences and were able to classify all APN permutations with linear self-equivalences in dimension $n = 6$.

1.1 Our Contribution

By adapting the algorithm of [BBL21], we find many new instances of quadratic APN functions. The search strategy is conceptually very simple. The basic idea is to fix the look-up table of the APN function F entry by entry. Each time a new entry is fixed, besides checking whether there is a

¹In 2021, after the preprint of our work was made public, Yu and Perrin [YP21] found more than 5,400 other new quadratic APN instances in dimension eight by the same method as in [YWL14], bringing up the total number of known 8-bit APN instances (including our results) to over 26,500. All of the APN instances found in [YP21] have an extended Walsh spectrum that was already known prior to our work (i.e., \mathcal{W}_0 , \mathcal{W}_1 , or \mathcal{W}_2 as defined in Section 4.2).

contradiction to the property of being APN, the algorithm further checks whether the values in the look-up table already imply the existence of a monomial of algebraic degree higher than two in the algebraic normal form of F . The main difference to the algorithm of [BBL21] is the incorporation of the check for the existence of high-degree monomials and the removal of the restriction on the bijectivity of the output function.

By using this approach, also combined with considering linear self-equivalences, we find 12,733 new CCZ-inequivalent quadratic APN functions in dimension $n = 8$, which is a substantial increase compared to the 8,192 previously-known APN instances in dimension $n = 8$.

Searching for new instances of APN functions in dimensions higher than $n = 8$ is known to be very hard and resource consuming. Indeed, to the best of our knowledge, previous search methods have not been very successful in finding new APN instances and the only previously-known APN instances in dimension $n \in \{9, 10\}$ are either monomial functions, one of the polynomials with coefficients in \mathbb{F}_2 [YKBL20], or those that come from the infinite families given in [BCL09b], [BCC⁺20], [BCC⁺21], [BC08], [BCL09a], [Tan19], or [BHK20]. Clearly, our approach becomes less efficient as well. However, we are still able to present 35 new APN instances in dimension $n = 9$ and five new APN instances in dimension $n = 10$. Remarkably, two of the new 9-bit APN functions are permutations. Until now, the only known APN permutations up to CCZ-equivalence were the monomial functions in odd dimension n , the binomial family for $3 \mid n$ presented in [BCL08] and the sporadic 6-bit APN permutation found in [BDMW10]. Up to EA-equivalence, the two new APN permutations can be given in univariate representation over \mathbb{F}_{2^9} by

$$\begin{aligned} x &\mapsto x^3 + ux^{10} + u^2x^{17} + u^4x^{80} + u^5x^{192}, \\ x &\mapsto x^3 + u^2x^{10} + ux^{24} + u^4x^{80} + u^6x^{136}, \end{aligned}$$

where $u \in \mathbb{F}_{2^9}^*$ is an element with minimal polynomial $X^3 + X + 1 \in \mathbb{F}_2[X]$.

Among the extended Walsh spectra of the APN functions found for $n = 8$, there are three that do not correspond to any of the previously-known quadratic 8-bit APN functions. In particular, there are four pairwise CCZ-inequivalent APN functions in dimension 8 having linearity 2^7 . One such example is the APN function

$$\begin{aligned} x &\mapsto x^3 + g^{60}x^5 + g^{191}x^6 + g^{198}x^9 + g^{232}x^{10} + g^{120}x^{12} + g^{54}x^{17} + g^{64}x^{18} + g^{159}x^{20} + \\ &g^{144}x^{24} + g^{248}x^{33} + g^{203}x^{34} + g^{32}x^{36} + g^{18}x^{40} + g^{216}x^{48} + g^{78}x^{65} + g^{46}x^{66} + g^{91}x^{68} + \\ &g^{27}x^{72} + g^{70}x^{80} + g^{52}x^{96} + g^{224}x^{129} + g^{18}x^{130} + g^{197}x^{136} + g^{253}x^{144} + x^{160} \end{aligned}$$

over \mathbb{F}_{2^8} , where $g \in \mathbb{F}_{2^8}^*$ is an element with minimal polynomial $X^8 + X^4 + X^3 + X^2 + 1 \in \mathbb{F}_2[X]$.

Finally, we apply the switching construction of Edel and Pott [EP09] to all the known (by the time of submission of this manuscript) and new quadratic APN instances in dimension $n = 7$ and $n = 8$, which leads to the discovery of another 188 CCZ-inequivalent APN instances in dimension $n = 8$. By using `sboxU` [Per17], we have checked that none of the APN functions we found for $n \in \{8, 10\}$ are CCZ-equivalent to a permutation.

The source code of our algorithms is publicly available at [cbe]. The look-up tables of the new APN instances are available in [BL21]. We emphasize that our search method is non-exhaustive, so we do not make any claim on the completeness of our findings.

2 Preliminaries

For a positive integer n , let us denote by \mathbb{F}_{2^n} the field with 2^n elements and let \mathbb{F}_2^n denote the n -dimensional vector space over \mathbb{F}_2 . Let $\mathbb{F}_{2^n}^*$ denote the set $\mathbb{F}_{2^n} \setminus \{0\}$. By $\text{GL}(n, \mathbb{F}_2)$ we denote the group of invertible $n \times n$ matrices over \mathbb{F}_2 and by $\text{AGL}(n, \mathbb{F}_2)$ we denote the group of affine permutations on \mathbb{F}_2^n . Any such affine permutation can be represented as $x \mapsto Lx + b$ for $L \in \text{GL}(n, \mathbb{F}_2)$ and $b \in \mathbb{F}_2^n$. To simplify notation, we are going to use elements of $\text{GL}(n, \mathbb{F}_2)$ and the linear functions that they represent interchangeably throughout this work. In other words, for an element $L \in \text{GL}(n, \mathbb{F}_2)$, we denote the linear function $x \mapsto Lx$ by L as well. The symbol I_n denotes the identity matrix in $\text{GL}(n, \mathbb{F}_2)$. By $\text{diag}(M_1, M_2, \dots, M_k)$, we denote the block-diagonal matrix consisting of the k blocks M_1, \dots, M_k , where M_1 corresponds to the upper-left block.

For an element $M \in \text{GL}(n, \mathbb{F}_2)$, we denote by $\text{ord}(M)$ the *multiplicative order* of M , which is defined as the smallest positive integer i such that $M^i = I_n$. Similarly, for $x \in \mathbb{F}_2^n$, we denote by $\text{ord}_M(x)$ the smallest positive integer i for which $M^i(x) = x$. The *minimal polynomial* of a matrix M over \mathbb{F}_2 is defined as the polynomial $p \in \mathbb{F}_2[X]$ of least positive degree such that $p(M) = 0$.

For a polynomial $q = X^n + q_{n-1}X^{n-1} + \dots + q_1X + q_0 \in \mathbb{F}_2[X]$, the *companion matrix* of q is defined as the $n \times n$ matrix

$$\text{Comp}(q) := \begin{bmatrix} 0 & & & & q_0 \\ 1 & 0 & & & q_1 \\ & \ddots & \ddots & & \vdots \\ & & & 1 & 0 & q_{n-2} \\ & & & & 1 & q_{n-1} \end{bmatrix},$$

which is an element of $\text{GL}(n, \mathbb{F}_2)$ if and only if $q_0 = 1$. For $x \in \mathbb{F}_2^n$, we denote by $\text{wt}(x)$ the Hamming weight of x , which is defined as the number of non-zero coordinates of x .

2.1 Representations of Vectorial Boolean Functions

For a comprehensive introduction to (vectorial) Boolean functions, we refer to [Car21]. Here, we recall the most important concepts needed in the remainder of the paper. Note that in the following, we restrict to the case of functions from \mathbb{F}_2^n to itself.

A vectorial Boolean function $F: \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ can be uniquely expressed as a multivariate polynomial in $\mathbb{F}_2^n[X_1, \dots, X_n]/(X_1^2 + X_1, \dots, X_n^2 + X_n)$, called the *algebraic normal form (ANF)*. In particular, there exist $a_u \in \mathbb{F}_2^n$ such that

$$F(x_1, \dots, x_n) = \sum_{u \in \mathbb{F}_2^n} \left(a_u \prod_{i \in \{1, \dots, n\}} x_i^{u_i} \right).$$

The *algebraic degree* of F is defined as $\max\{\text{wt}(u) \mid a_u \neq 0, u \in \mathbb{F}_2^n\}$. The function F is called *affine* if it is of algebraic degree at most 1 and it is called *quadratic* if it is of algebraic degree 2. The coefficients a_u of the ANF can be obtained by the so-called binary Möbius transform via

$$a_u = \sum_{x \in \mathbb{F}_2^n, x \preceq u} F(x), \tag{1}$$

where the relation $x \preceq u$ holds if and only if, for all $i \in \{1, \dots, n\}$, we have $(u_i = 0 \Rightarrow x_i = 0)$.

Moreover, we can uniquely represent any vectorial Boolean function $F: \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ as a function from \mathbb{F}_{2^n} to itself via $x \mapsto f(x)$ with $f \in \mathbb{F}_{2^n}[X]/(X^{2^n} + X)$. This is called the *univariate representation* of F .

The *Walsh transform* of F at $(\alpha, \beta) \in \mathbb{F}_2^n \times \mathbb{F}_2^n$ is defined as the sum

$$\widehat{F}(\alpha, \beta) = \sum_{x \in \mathbb{F}_2^n} (-1)^{\langle \alpha, x \rangle + \langle \beta, F(x) \rangle}$$

over the integers, where $\langle x, y \rangle$ denotes the inner product of the vectors $x, y \in \mathbb{F}_2^n$, defined as $\langle x, y \rangle := \sum_{i=1}^n x_i y_i \pmod{2}$. The multiset $\{|\widehat{F}(\alpha, \beta)| \mid \alpha, \beta \in \mathbb{F}_2^n\}$ is called the *extended Walsh spectrum* of F . For $\beta \in \mathbb{F}_2^n \setminus \{0\}$, the function $F_\beta: x \mapsto \langle \beta, F(x) \rangle$ is called a *component* of F and the value $\max_{\alpha \in \mathbb{F}_2^n, \beta \in \mathbb{F}_2^n \setminus \{0\}} |\widehat{F}(\alpha, \beta)|$ is called the *linearity* of F . The linearity of F can be understood as a measure of how well a component of F can be approximated by an affine function. In particular, a linearity of 2^n corresponds to the case of F having an affine component.

The *difference distribution table (DDT)* of a function $F: \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ is the $2^n \times 2^n$ integer matrix (where the rows and the columns are indexed by $\alpha \in \mathbb{F}_2^n$ and $\beta \in \mathbb{F}_2^n$, respectively) that contains $|\{x \in \mathbb{F}_2^n \mid F(x) + F(x + \alpha) = \beta\}|$ in the entry in row α and column β . The *differential spectrum* of F is defined as the multiset of entries in the DDT of F .

2.2 Equivalence Relations of Vectorial Boolean Functions

Let $F, G: \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$. There are several well-known equivalence relations on vectorial Boolean functions that preserve both the differential spectrum and the extended Walsh spectrum. The function G is *linear-equivalent* to F if there exist $A, B \in \text{GL}(n, \mathbb{F}_2)$ such that $F \circ A = B \circ G$. Moreover, G is *extended affine-equivalent (EA-equivalent)* to F if there exist $A, B \in \text{AGL}(n, \mathbb{F}_2)$ and an affine, not necessarily invertible, function $C: \mathbb{F}_2^n \mapsto \mathbb{F}_2^n$ such that $F \circ A = B \circ G + C$. The functions F and G are called *CCZ-equivalent* [BCP06, CCZ98] if there exists a transformation $\sigma \in \text{AGL}(2n, \mathbb{F}_2)$ such that $\Gamma_G = \sigma(\Gamma_F)$, where $\Gamma_F := \{(x, F(x)) \mid x \in \mathbb{F}_2^n\}$ denotes the graph of F . Among the notions of equivalence listed above, CCZ-equivalence is the most general. An important goal in the study of APN functions is to classify them up to CCZ-equivalence and the following is a useful result for quadratic APN functions.

Theorem 1 ([Yos12]). *Two quadratic APN functions $F, G: \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ are EA-equivalent if and only if they are CCZ-equivalent.*

Therefore, since our focus is on quadratic APN functions, we are going to separate the functions we find up to EA-equivalence (which corresponds to CCZ-equivalence in this case) and we only provide a single representative from each CCZ-equivalence class. Such a CCZ-equivalence class will also be called an *instance* throughout this paper, which is represented by one member of the class.

The *LE-automorphism* group (see [BBL21]) of a function $F: \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ is defined as

$$\text{Aut}_{\text{LE}}(F) := \{\text{diag}(A, B) \in \text{GL}(2n, \mathbb{F}_2) \mid A, B \in \text{GL}(n, \mathbb{F}_2) \text{ and } F \circ A = B \circ F\}.$$

If a function F admits a non-trivial automorphism $\text{diag}(A, B) \in \text{Aut}_{\text{LE}}(F)$, we also say that F is *linearly self-equivalent* with respect to the tuple (B, A) . Note that if $F: \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ and $G: \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ are linear-equivalent, we have $\text{Aut}_{\text{LE}}(F) \cong \text{Aut}_{\text{LE}}(G)$, where \cong denotes the isomorphism relation of groups.

3 A Recursive Tree Search for Quadratic APN Functions

Our idea for finding new instances of APN functions is to apply a recursive tree search very similar to [BBL21, Algorithm 1]. The main differences are that in the algorithm given here we drop the restriction to only search for permutations and we apply an additional filter for skipping branches within the search tree that lead to functions of algebraic degree higher than two. In this section, we explain the most basic variant that searches for arbitrary n -bit quadratic APN functions for $n \geq 2$. It is formally specified in Algorithm 1. The algorithm can easily be adapted to only search for quadratic APN functions admitting a particular LE-automorphism (see Section 4.2).

The global array `sbox` is initialized to be undefined (\perp) at each entry. This array corresponds to the look-up table of the APN function F to be constructed. In each iteration of `NEXTVAL`, the procedure `ISCOMPLETE` first checks whether `sbox` is already completely defined. If this is the case, the algorithm has found a quadratic APN function and prints `sbox` as the solution. Otherwise, the procedure `NEXTFREEPOSITION` is called which selects the next undefined entry x and sets $F(x)$ to a value y that is randomly selected from among a predefined list of possible choices. The orders in which those values y are selected at each depth are determined by the `SHUFFLE` procedure that is performed in the beginning. After fixing $F(x)$, the procedure `ADDPPOINT` checks whether F can still be both APN and quadratic. If not, the current branch of the search tree is skipped and x is set to the next possible value y . In case no contradiction to either property is encountered, the algorithm goes one level deeper.

Note that, since the running time of Algorithm 1 can be very long in cases where no quadratic APN function is found, we abort and restart after a predetermined amount of time (e.g., 10 seconds for $n = 7$).

3.1 APN Check

Each time we set $F(x)$ to a new value y , we need to check that the APN property of F has not already been violated. This is performed in exactly the same way as in [BBL21]. In particular, the function `ADDDDTINFORMATION(x)` dynamically changes the DDT according to the value set by the current iteration. The DDT is stored in a global array which is initialized to 0 before calling Algorithm 1. Similarly, each time we reset $F(x)$ to \perp , the function `REMOVEDDTINFORMATION(x)` applies the appropriate changes to the DDT. Note that `ADDDDTINFORMATION(x)` returns 1 if the APN property is not violated by fixing $F(x)$. Otherwise, it returns 0. Similarly, `REMOVEDDTINFORMATION(x)` returns 1 if the APN property has not been violated by fixing $F(x)$. Otherwise, it returns 0.

3.2 Algebraic Degree Check

Each time we set $F(x)$ to a new value y , we check whether we can deduce the existence of a monomial of algebraic degree higher than 2 in the algebraic normal form of F . For this, using Equation (1), we keep track of the partial sums for all a_u with $\text{wt}(u) \geq 3$ in a global array `sum` and update them whenever $x \preceq u$. The check is performed by calling the function `ADDDEGREEINFORMATION(x)`, which is defined below. The function dynamically changes the global arrays `ctr` and `sum`, both of size 2^n , which are initialized to 0 before calling Algorithm 1. Below, the symbol \oplus denotes the bitwise XOR operation in order to distinguish it from the addition of integers, denoted $+$.

- 1: **function** `ADDDEGREEINFORMATION(x)`
- 2: **for** $u \in [1, \dots, 2^n - 1]$ such that $\text{wt}(u) \geq 3$ and $x \preceq u$ **do**

```

3:     ctr[u] ← ctr[u] + 1
4:     sum[u] ← sum[u] ⊕ sbox[x]
5:     if ctr[u] = 2wt(u) then                                ▷ All x with x ≼ u have been considered
6:         if sum[u] ≠ 0 then                                    ▷ We have a_u ≠ 0 in the ANF of F
7:             return 0
8:         end if
9:     end if
10: end for
11: return 1
12: end function

```

When $F(x)$ is reset to \perp , we need to restore the values of the arrays `ctr` and `sum` by calling the following procedure.

```

1: function REMOVEDEGREEINFORMATION(x)
2:   for u ∈ [1, ..., 2n - 1] such that wt(u) ≥ 3 and x ≼ u do
3:     ctr[u] ← ctr[u] - 1
4:     sum[u] ← sum[u] ⊕ sbox[x]
5:     if ctr[u] = 2wt(u) - 1 then
6:         if sum[u] ≠ sbox[x] then
7:             return 0
8:         end if
9:     end if
10:  end for
11:  return 1
12: end function

```

3.3 EA-equivalence Check

For each function that we find, we need to check whether it is EA-equivalent to a known instance. We recall that for two quadratic APN functions, EA-equivalence coincides with CCZ-equivalence. To perform the check efficiently, we use the following method by Canteaut et al., first explained in an invited talk at Boolean Functions and their Applications (BFA) 2020 and formally described in the recent preprint [CCP21].

Proposition 1 ([CP20, CCP21]). *Let $F: \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ be a quadratic APN function. The ortho-derivative of F is defined as the unique function $\Pi_F: \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ with $\Pi_F(0) = 0$ such that, for all $\alpha \in \mathbb{F}_2^n \setminus \{0\}$, we have $\Pi_F(\alpha) \neq 0$ and*

$$\forall x \in \mathbb{F}_2^n: \langle \Pi_F(\alpha), (F(x) + F(x + \alpha) + F(\alpha) + F(0)) \rangle = 0.$$

For two EA-equivalent quadratic APN functions $F, G: \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$, the ortho-derivatives Π_F and Π_G are linear-equivalent.

Testing two quadratic APN functions for EA-inequivalence (which is the same as CCZ-inequivalence in this case) is now fairly simple. One simply computes the corresponding ortho-derivatives and evaluates their extended Walsh spectra and differential spectra. We note that these are (strongly discriminating) invariants for EA-equivalence. If the two extended Walsh spectra or the two differential spectra are different, the ortho-derivatives cannot be linear-equivalent and hence, the two

Algorithm 1 QUADRATICAPNSEARCH

Input: Global array sbox of size 2^n , initialized to $\text{sbox}[i] = \perp$, for all $i \in \{0, \dots, 2^n - 1\}$. Global 2-dimensional array P of size $2^n \times 2^n$ with each $P[i]$ initialized to $[0, \dots, 2^n - 1]$ for all $i \in \{0, \dots, 2^n - 1\}$.

Output: Prints an n -bit APN function F of algebraic degree at most two.

```
1: for  $i \in [0, \dots, 2^n - 1]$  do
2:   SHUFFLE( $P[i]$ ) ▷ Generates a random permutation of  $P[i]$ 
3: end for
4:  $\text{sbox}[0] \leftarrow 0$ 
5: ADDPOINT(0)
6: NEXTVAL(0)

7: function NEXTVAL(depth)
8:   if ISCOMPLETE( $\text{sbox}$ ) then ▷ Checks if  $\text{sbox}$  contains no  $\perp$ 
9:     Print  $\text{sbox}$  and terminate
10:  end if
11:   $x \leftarrow$  NEXTFREEPOSITION() ▷ Chooses the smallest  $i$  s.t.  $\text{sbox}[i] \neq \perp$ 
12:  for  $z \in [0, \dots, 2^n - 1]$  do
13:     $y \leftarrow P[\text{depth}][z]$ 
14:     $\text{sbox}[x] \leftarrow y$ 
15:     $b \leftarrow$  ADDPOINT( $x$ )
16:    if  $b$  is equal to 1 then
17:      NEXTVAL(depth + 1)
18:    end if
19:     $\text{sbox}[x] \leftarrow \perp$ 
20:    REMOVEPOINT( $x$ )
21:  end for
22: end function

23: function ADDPOINT( $c$ )
24:   if ADDDDTINFORMATION( $c$ ) then
25:     return ADDDEGREEINFORMATION( $c$ )
26:   end if
27:   return 0
28: end function

29: function REMOVEPOINT( $c$ )
30:   if REMOVEDDTINFORMATION( $c$ ) then
31:     REMOVEDEGREEINFORMATION( $c$ )
32:   end if
33: end function
```

quadratic APN functions cannot be EA-equivalent. The implementation for computing the ortho-derivative is contained in the latest version of `sboxU` [Per17]. This method is much more efficient than checking the code equivalence with Magma [BCP97].

It might be the case that two EA-inequivalent functions are not identified as such because their ortho-derivatives might have identical differential and extended Walsh spectra. However, this does not seem to occur often. For example, all of the previously-known 8,191 quadratic 8-bit APN instances can be established to be EA-inequivalent by this method. The complete check for all of those 8,191 APN instances only takes a few minutes on a PC.

We remark that no further effort is needed to test the quadratic APN functions that we find for CCZ-inequivalence to the non-quadratic monomial APN functions. This is because of the fact that a quadratic APN function CCZ-equivalent to a monomial function must be EA-equivalent to a quadratic monomial function [Yos16].

3.4 Results

After running the search for $n = 7$ for about 72 CPU hours, we found most of the quadratic APN instances, including the recently discovered APN function presented in [KI20b]. For higher values of n , this direct approach is not very efficient and so we consider linear self-equivalences in the following.

4 Considering LE-Automorphisms

We now describe the method for searching for quadratic APN functions with non-trivial LE-automorphisms in small dimension n and apply it to $n \in \{7, 8, 9, 10\}$. Again, the algorithm is similar to the one presented in [BBL21] where the focus was on APN permutations of arbitrary algebraic degree.

4.1 Canonical classes of LE-Automorphisms

If we consider n -bit functions F with non-trivial elements in $\text{Aut}_{\text{LE}}(F)$ and are only interested in a classification of such F up to linear-equivalence,² we can significantly reduce the number of matrix pairs (B, A) that we need to consider. The following ideas and reductions have already been presented in [BBL21] with a focus on the case of F being a permutation.

Lemma 1 ([BBL21]). *Let $F: \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ have a non-trivial automorphism in $\text{Aut}_{\text{LE}}(F)$. There exist $A, B \in \text{GL}(n, \mathbb{F}_2)$ with $F \circ A = B \circ F$ such that either*

1. $\text{ord}(A) = \text{ord}(B) = p$ for p prime, or
2. $B = I_n$ and $\text{ord}(A) = p$ for p prime, or
3. $A = I_n$ and $\text{ord}(B) = p$ for p prime.

Two elements $M, M' \in \text{GL}(n, \mathbb{F}_2)$ are called *similar*, denoted $M \sim M'$, if there exists $P \in \text{GL}(n, \mathbb{F}_2)$ such that $M' = P^{-1}MP$. Similarity is an equivalence relation on $\text{GL}(n, \mathbb{F}_2)$ and we can find a representative of each equivalence class by the so-called rational canonical form.

²We cannot use EA-equivalence here because the property of admitting a non-trivial LE-automorphism is not invariant under EA-equivalence.

Lemma 2. (*Rational Canonical Form*)[DF04, Page 476] Every element $M \in \text{GL}(n, \mathbb{F}_2)$ is similar to a unique $M' \in \text{GL}(n, \mathbb{F}_2)$ of the form $M' = \text{diag}(\text{Comp}(q_r), \text{Comp}(q_{r-1}), \dots, \text{Comp}(q_1))$ for polynomials q_i such that $q_r \mid q_{r-1} \mid \dots \mid q_1$. This matrix M' is called the rational canonical form of M , denoted $\text{RCF}(M)$.

If we want to collect all LE-automorphisms in order to classify all functions $F: \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ that admit a non-trivial linear self-equivalence up to linear-equivalence, we can without loss of generality assume that A and B are in rational canonical form. The following definition and lemma allow us to reduce the search space even further.

Definition 2 ([BBL21]). Let $A, B, C, D \in \text{GL}(n, \mathbb{F}_2)$ be of order p for p prime. The tuple (A, B) is said to be power-similar to the tuple (C, D) , denoted $(A, B) \sim_p (C, D)$, if there exists a positive integer i such that $A \sim C^i$ and $B \sim D^i$.

Power-similarity defines an equivalence relation on the ordered pairs of matrices in $\text{GL}(n, \mathbb{F}_2^n)$ of the same prime order and the following holds.

Lemma 3 ([BBL21]). Let $F: \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ with $\text{diag}(A, B) \in \text{Aut}_{\text{LE}}(F)$ for $A, B \in \text{GL}(n, \mathbb{F}_2)$ being of prime order p . For every (\tilde{B}, \tilde{A}) power-similar to (B, A) , there is a function G linear-equivalent to F such that $\text{diag}(\text{RCF}(\tilde{A}), \text{RCF}(\tilde{B})) \in \text{Aut}_{\text{LE}}(G)$.

For the dimensions that we consider in this work, i.e., $n \leq 10$, we can efficiently generate all rational canonical forms as block-diagonal matrices of the form given in Lemma 2. By applying the above Lemmas 1 and 3 to a fixed dimension n , we obtain a reduced number of tuples (B, A) to consider. We call those *canonical classes* of LE-automorphisms.

$n = 7$ All linear-equivalence classes of functions $F: \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ that admit a non-trivial linear self-equivalence can be obtained by considering 128 canonical classes for pairs (B, A) : 56 classes with $\text{ord}(A) = \text{ord}(B)$ being prime, 36 classes with $B = I_n$ and $\text{ord}(A)$ being prime, and 36 classes with $A = I_n$ and $\text{ord}(B)$ being prime.

$n = 8$ All linear-equivalence classes of functions $F: \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ that admit a non-trivial linear self-equivalence can be obtained by considering 157 canonical classes for pairs (B, A) : 75 classes with $\text{ord}(A) = \text{ord}(B)$ being prime, 41 classes with $B = I_n$ and $\text{ord}(A)$ being prime, and 41 classes with $A = I_n$ and $\text{ord}(B)$ being prime.

$n = 9$ All linear-equivalence classes of functions $F: \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ that admit a non-trivial linear self-equivalence can be obtained by considering 217 canonical classes for pairs (B, A) : 111 classes with $\text{ord}(A) = \text{ord}(B)$ being prime, 53 classes with $B = I_n$ and $\text{ord}(A)$ being prime, and 53 classes with $A = I_n$ and $\text{ord}(B)$ being prime.

$n = 10$ All linear-equivalence classes of functions $F: \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ that admit a non-trivial linear self-equivalence can be obtained by considering 401 canonical classes for pairs (B, A) : 247 classes with $\text{ord}(A) = \text{ord}(B)$ being prime, 77 classes with $B = I_n$ and $\text{ord}(A)$ being prime, and 77 classes with $A = I_n$ and $\text{ord}(B)$ being prime.

The tuples (B, A) and their corresponding class indices can be found along with our source code at [cbe].

4.2 Finding quadratic APN functions in the canonical classes

If we are searching for APN functions (not necessarily quadratic) with non-trivial LE-automorphisms, we do not have to check all of the canonical classes, as outlined in the following two lemmas. For a matrix $M \in \text{GL}(n, \mathbb{F}_2)$, let us denote by $\text{Fix}_M := \{x \in \mathbb{F}_2^n \mid Mx = x\}$ the set of fixed points of M , which is a linear subspace of \mathbb{F}_2^n .

Lemma 4. *Let $A, B \in \text{GL}(n, \mathbb{F}_2)$. If B has strictly less fixed points than A and $(|\text{Fix}_B|, |\text{Fix}_A|) \notin \{(1, 2), (2, 4)\}$, no function $F: \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ with $F \circ A = B \circ F$ can be APN.*

Proof. Let F be such that $F \circ A = B \circ F$. For all $x \in \text{Fix}_A$, we have $F(x) = B(F(x))$, thus $\{F(x) \mid x \in \text{Fix}_A\} \subseteq \text{Fix}_B$. Since Fix_A and Fix_B are linear subspaces of \mathbb{F}_2^n , if $|\text{Fix}_B| < |\text{Fix}_A|$, the image of the restriction of F on the subspace Fix_A (with $\dim \text{Fix}_A = k$) is contained in a subspace of smaller dimension $\ell < k$, where $\ell = \dim \text{Fix}_B$. If F is APN, this would imply the existence of an APN function H on \mathbb{F}_2^k whose image set is contained in $\mathbb{F}_2^\ell \times \{0\}^{k-\ell}$. The APN function H must therefore have a component constant and equal to zero, more precisely, it must have $2^{k-\ell} - 1$ of them. According to [Car21, Proposition 161], this is not possible, unless $k \leq 2$. Since the cases of $(\ell, k) \in \{(0, 1), (1, 2)\}$ are excluded in the statement of the lemma, we only need to consider the case of $(\ell, k) = (0, 2)$. In this case, we have that $H: \mathbb{F}_2^2 \rightarrow \mathbb{F}_2^2$ is constant and equal to zero, so H is not an APN function. \square

The following lemma is a direct consequence of [CHP17, Lemma 5] and states that classes of the form (I_n, A) only need to be considered if the cycle decomposition of A (where A is considered as a permutation on the set \mathbb{F}_2^n) consists of a large number of cycles.

Lemma 5. *Let $A \in \text{GL}(n, \mathbb{F}_2)$ and let $F: \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ be an APN function with $F \circ A = F$. Let us denote by c be the number of distinct cycles of A . If n is even, we have $c \geq \frac{2^n+2}{3}$, and, if n is odd, we have $c \geq \frac{2^n+1}{3}$.*

Proof. For an n -bit function F with $F \circ A = F$, we have for the image size $|\{F(x) \mid x \in \mathbb{F}_2^n\}| \leq c$. From [CHP17, Lemma 5] (see also [Car20] for a more recent discussion of this result), if F is APN, we have

$$c \geq \left\lceil \frac{2^{2n}}{3 \cdot 2^n - 2} \right\rceil,$$

where the term on the right-hand side is equal to $\frac{2^n+2}{3}$ if n is even and $\frac{2^n+1}{3}$ if n is odd. \square

For the remaining (non-trivial) canonical classes of LE-automorphisms in dimensions $n = 7$ and $n = 8$ not excluded by Lemmas 4 and 5 (i.e., 53 classes for $n = 7$ and 67 classes for $n = 8$) and for most of³ the remaining (non-trivial) canonical classes of LE-automorphisms in dimensions $n = 9$ and $n = 10$, we performed a randomized tree search for quadratic APN functions similar to the search described in Algorithm 1. The main difference is that, once we fix an element $F(x) = y$, we fix $F(A^i(x)) = B^i(y)$, $i \in \{1, \dots, \text{ord}_A(x) - 1\}$ as well. Moreover, if A and B have an identical number of fixed points, we set the restriction of F on Fix_A to an APN function before invoking the recursion. More precisely, let F be a quadratic APN function with $F \circ A = B \circ F$, where $\dim \text{Fix}_A = \dim \text{Fix}_B = k$. Let further $\pi_A: \mathbb{F}_2^k \rightarrow \text{Fix}_A$ and $\pi_B: \mathbb{F}_2^k \rightarrow \text{Fix}_B$ be (vector space) isomorphisms. Then, there exists an APN function $G: \mathbb{F}_2^k \rightarrow \mathbb{F}_2^k$ of algebraic degree at most two

³We did not perform any search if $|\text{Fix}_A| \geq 2^8$. For this reason, we excluded 2 and 7 of the remaining (non-trivial) canonical classes of LE-automorphisms in the search in dimension $n = 9$ and $n = 10$, respectively.

such that, for all $x \in \mathbb{F}_2^k$, we have $F(\pi_A(x)) = \pi_B(G(x))$. It is not clear whether we can choose G up to EA-equivalence. However, in our randomized search, we only select G from a predetermined list of EA-representatives. For example, if $k = 6$, we choose one of 13 quadratic APN instances uniformly at random. We refer to our source code for the lists of EA-representatives from which we choose for each fixed k . Note that our algorithm might terminate without returning a quadratic APN function although one exists. This is the case if the chosen G does not yield a quadratic n -bit APN function, while another choice of G would. However, we do not know whether this can happen.

Special cases There are some special cases of classes for which we slightly modify the algorithm. Those cases are the LE-automorphisms for which $1 < |\text{Fix}_A| < |\text{Fix}_B|$. If $2 = |\text{Fix}_A| < |\text{Fix}_B|$ with $\text{Fix}_A = \{0, x\}$, we set $F(x) = 0$ in the beginning before calling `NEXTVAL(0)`. However, if $2 < |\text{Fix}_A| < |\text{Fix}_B|$, we do not set any points in the beginning besides $F(0) = 0$. Instead, we first select those positions x for which $\text{ord}_A(x) > 1$ in `NEXTFREEPOSITION()` and the fixed points of A last. This allows us to fix large cycles first.

Deterministic search It is possible to slightly modify the randomized algorithm to perform a deterministic search. In particular, we run the search for all possible choices of k -bit APN instances G of algebraic degree at most two and do not abort after a predetermined amount of time or if a solution is found. Similarly as it was done in the exhaustive search presented in [BBL21], we call `NEXTVAL(depth + 1)` only if F is the smallest representative (up to some previously-defined ordering) in the set $\{C_B \circ F \circ C_A \mid C_A \in \mathcal{C}_A, C_B \in \mathcal{C}_B\}$, where \mathcal{C}_A (resp., \mathcal{C}_B) is a subset of all elements in $\text{GL}(n, \mathbb{F}_2)$ that commute with A (resp., B) and whose restriction on Fix_A (resp., Fix_B) is the identity. Formally, $\mathcal{C}_A \subseteq \{M \in \text{GL}(n, \mathbb{F}_2) \mid MA = AM \text{ and } \forall x \in \text{Fix}_A : Mx = x\}$ (similarly for \mathcal{C}_B). Note that this method does not necessarily result in an exhaustive search as it is not clear whether we can select G up to EA-equivalence.

For each canonical class of LE-automorphisms, before starting the randomized search, we first check whether the deterministic search terminates in short time (i.e., a few minutes to hours). If this is the case, we do not invoke the randomized search.

Results for $n = 7$ The only APN functions found are those which are EA-equivalent to univariate polynomials with coefficients in \mathbb{F}_2 . Quadratic APN polynomials with coefficients in \mathbb{F}_2 have already been classified for $n = 7$, see [YKBL20].

Results for $n = 8$ To the best of our knowledge, by the time of submission of this manuscript in December 2020 and excluding our results, there are 8,192 known instances of 8-bit APN functions, i.e., the 23 instances listed in [EP09], the 8,157 instances constructed by the QAM method [YWL13, YWL14], the 10 instances presented in [WTG13], and the two instances from the Taniguchi family [Tan19]. With our approach, we find 12,733 new instances of quadratic APN functions. In particular, we find quadratic APN functions within 9 different classes of LE-automorphisms (see Table 1). For each canonical class of LE-automorphisms that could not be directly excluded by Lemma 4 or Lemma 5, we performed the search for a few CPU days at most.⁴

The extended Walsh spectra of all of those 12,733 new instances belong to one of the six spectra $\mathcal{W}_0, \dots, \mathcal{W}_5$ listed below. By $a : m$, we indicate that the value a occurs with multiplicity m in

⁴Note that for a lot of classes, the search terminates immediately without solutions.

the multiset. APN functions in dimension $n = 8$ having $\mathcal{W}_0, \mathcal{W}_1$, or \mathcal{W}_2 as their extended Walsh spectrum are already known. There was no previously-known APN function with extended Walsh spectrum $\mathcal{W}_3, \mathcal{W}_4$, or \mathcal{W}_5 .

$$\begin{aligned}\mathcal{W}_0 &= \{0 : 16320, 16 : 43520, 32 : 5440\} \quad (\text{classical spectrum}) \\ \mathcal{W}_1 &= \{0 : 15600, 16 : 44544, 32 : 5120, 64 : 16\} \\ \mathcal{W}_2 &= \{0 : 14880, 16 : 45568, 32 : 4800, 64 : 32\} \\ \mathcal{W}_3 &= \{0 : 14160, 16 : 46592, 32 : 4480, 64 : 48\} \quad (\text{new}) \\ \mathcal{W}_4 &= \{0 : 13440, 16 : 47616, 32 : 4160, 64 : 64\} \quad (\text{new}) \\ \mathcal{W}_5 &= \{0 : 12540, 16 : 48640, 32 : 4096, 128 : 4\} \quad (\text{new})\end{aligned}$$

Certainly, \mathcal{W}_5 is the most interesting extended Walsh spectrum since it corresponds to 8-bit functions with linearity 2^7 and we found four such instances of quadratic APN functions. Whether quadratic n -bit APN functions with linearity 2^{n-1} exist was mentioned as an open problem in [Car18]. Before now, besides the trivial cases in dimension $n \leq 4$, we only knew one such instance in dimension $n = 6$, see [EP09].

Table 1: For all class indices (no.) in dimension 8 for which we find solutions, this table gives a lower bound on the number of distinct EA-equivalence classes of quadratic APN functions that admit the particular LE-automorphism, separated by their extended Walsh spectra. The numbers in parentheses indicate the number of instances that are not contained in the previously known 8,192 instances of APN functions.

no.	\mathcal{W}_0	\mathcal{W}_1	\mathcal{W}_2	\mathcal{W}_3	\mathcal{W}_4	\mathcal{W}_5
1	1 (0)					
2	1 (0)					
21	9 (9)					
31	7 (4)	1 (1)		2 (2)		
38	3 (0)					
51	24 (20)					
55	9,093 (9,090)	3,065 (3,065)	299 (297)	146 (146)	25 (25)	4 (4)
56	103 (79)		2 (2)		1 (1)	
113	26 (0)					

In Class 1 (up to a change of basis), the matrix A corresponds to multiplication by a non-zero field element $\alpha \in \mathbb{F}_{2^n}$ of multiplicative order 17 and the matrix B corresponds to multiplication by α^3 . Thus, as a solution, we find the APN function $x \mapsto x^3$. Similarly, in Class 2, the matrix A corresponds to multiplication by a non-zero field element $\alpha \in \mathbb{F}_{2^n}$ of multiplicative order 17 and the matrix B corresponds to multiplication by α^9 . Therefore, as a solution, we find the APN function $x \mapsto x^9$.

Class 56 corresponds to those functions F whose univariate representation only contains coefficients in the subfield \mathbb{F}_{2^4} .

Let $\zeta_3 \in \mathbb{F}_{2^8}^* \setminus \{1\}$ be a third root of unity. Class 113 corresponds to those functions F for which $F(x) = F(\zeta_3 x)$. Class 51 corresponds to the linear self-equivalence where A is the multiplication by

ζ_3 and B is similar to the block diagonal matrix $\text{diag}(I_2, \text{Comp}(X^3 + 1), \text{Comp}(X^3 + 1))$ and Class 55 corresponds to the linear self-equivalence where A is the multiplication by ζ_3 and B is similar to the block-diagonal matrix $\text{diag}(I_5, \text{Comp}(X^3 + 1))$. Note that in both cases, B does not correspond to multiplication by a finite field element or to a linear mapping of the form $x \mapsto x^{2^i}$.

Let $\zeta_5 \in \mathbb{F}_{2^8}^* \setminus \{1\}$ be a fifth root of unity. Class 38 corresponds to those functions F for which $\zeta_5(F(x)) = F(\zeta_5 x)$. Class 31 corresponds to the linear self-equivalence where A is the multiplication by ζ_5 and B is similar to the block-diagonal matrix $\text{diag}(I_3, \text{Comp}(X^5 + 1))$. Again, B does not correspond to multiplication by a finite field element or to a linear mapping of the form $x \mapsto x^{2^i}$.

Class 21 corresponds to the linear self-equivalence given by $A = B = \text{diag}(I_1, \text{Comp}(X^7 + 1))$.

Results for $n = 9$ To the best of our knowledge, by the time of submission, the only known APN instances for $n = 9$ either correspond to polynomials with coefficients in \mathbb{F}_2 [YKBL20], to the (generalized) isotopic shift construction [BCC⁺20, BCC⁺21], or to the infinite families given in [BCL09a, BCL09b]. The only APN *permutations* in dimension nine known so far are CCZ-equivalent to monomial functions. We applied our search for $n = 9$ and found 35 new APN instances, two of them being permutations. Up to linear-equivalence, those two instances of APN permutations F_1, F_2 admit a linear self-equivalence of the form $F_i(u^5 x) = u F_i(x)$, where $u \in \mathbb{F}_{2^3}^*$.

Functions fulfilling this self-equivalence can be characterized by the property that their univariate representation does not contain monomials x^j with $j \not\equiv 3 \pmod{7}$. Interestingly, this property is also fulfilled by the “Kim mapping”, i.e., the function

$$K: \mathbb{F}_{2^6} \rightarrow \mathbb{F}_{2^6}, \quad x \mapsto x^3 + x^{10} + gx^{24},$$

where g is an element in $\mathbb{F}_{2^6}^*$ with minimal polynomial $X^6 + X^4 + X^3 + X + 1 \in \mathbb{F}_2[X]$. It was shown in [BDMW10] that K is CCZ-equivalent to an APN permutation.

Results for $n = 10$ To the best of our knowledge, by the time of submission, the only known APN instances for $n = 10$ are either monomial functions or those that come from the infinite families given in [BC08], [BCL09a], [Tan19], and [BHK20] (see the instances 10.1–10.17 in the list available at https://boolean.h.uib.no/mediawiki/index.php/CCZ-inequivalent_representatives_from_the_known_APN_families_for_dimensions_up_to_11).⁵ We applied our approach for $n = 10$ and found 5 APN instances that are CCZ-inequivalent to the known 17 instances that come from infinite families.

The look-up tables of all of the new APN instances that we found in dimension $n \in \{8, 9, 10\}$ are available in [BL21].

5 Further APN Instances from the Switching Construction

In [EP09], the authors presented the switching construction which potentially allows to generate CCZ-inequivalent APN functions from a given APN function by replacing one of its components. Using this method, they found the only known APN instance that is not CCZ-equivalent to a monomial function or to a quadratic function.

⁵accessed October 6, 2021. As noted in the disclaimer, not all instances of the family given in [BCC⁺20] could have been checked. Therefore, it could be possible that our instances are coming from that family.

Definition 3. Two functions $F, G: \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ are said to be switching neighbours if there exists a Boolean function $f: \mathbb{F}_2^n \rightarrow \mathbb{F}_2$ and a non-zero vector $v \in \mathbb{F}_2^n$ such that, for all $x \in \mathbb{F}_2^n$, we have $G(x) = F(x) + vf(x)$.

It has been shown in [EP09, Theorem 3] that, if $F: \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ is an APN function, all APN functions that are switching neighbours of F can be found by simple linear algebra. Indeed, for a non-zero vector $v \in \mathbb{F}_2^n$, to find all functions $f: \mathbb{F}_2^n \rightarrow \mathbb{F}_2$ such that $F + vf$ is an APN function, one can collect all tuples $(x, y, x + y, a), x, y, a \in \mathbb{F}_2^n$ for which $F(x) + F(x + a) + F(y) + F(y + a) = v$ holds and for each such tuple include the linear equation

$$f_x + f_{x+a} + f_y + f_{y+a} = 0$$

to the system to solve, where the unknowns are $f_z \in \mathbb{F}_2$ for $z \in \mathbb{F}_2^n$. Any solution $(f_z)_{z \in \mathbb{F}_2^n}$ of this linear system corresponds to a function f with $f(z) = f_z, z \in \mathbb{F}_2^n$ such that $F + vf$ is APN.

We applied this method to all of the known (by the time of submission of this manuscript) and new APN instances for $n = 7$ and $n = 8$. For $n = 8$, we found 188 new APN instances in this way. All of those functions are quadratic. The look-up tables of those 188 functions are also available in the dataset [BL21].

6 Conclusion

We performed a recursive search for quadratic APN functions in small dimension and we have shown that quadratic APN functions with linearity 2^{n-1} exist for the case of $n = 8$. Further, we found two previously unknown APN permutations in dimension $n = 9$. An open question is whether APN functions with linearity 2^{n-1} also exist in any even dimension $n > 8$. To answer this question, it would be interesting to generalize one of the four eight-bit APN functions with high linearity presented in this work to an infinite family. Another question is whether the list of known extended Walsh spectra of quadratic APN functions in dimension $n = 8$ is complete. Finally, the problem of classifying the new APN permutations in dimension $n = 9$ into infinite families remains open. In this direction, it would be interesting to analyze similarities between the Kim function and the two new APN permutations.

Acknowledgment

We thank the associate editor and the anonymous reviewers for their detailed and helpful comments. We further thank Léo Perrin for pointing us to the idea of using the ortho-derivative for checking EA-equivalence of our found functions.

References

- [Ars18] R. Arshad. *Contributions to the theory of almost perfect nonlinear functions*. PhD thesis, Otto-von-Guericke-Universität Magdeburg, Fakultät für Mathematik, 2018.
- [BBL21] C. Beierle, M. Brinkmann, and G. Leander. Linearly self-equivalent APN permutations in small dimension. *IEEE Trans. Inf. Theory*, 67(7):4863–4875, 2021.

- [BC08] L. Budaghyan and C. Carlet. Classes of quadratic APN trinomials and hexanomials and related structures. *IEEE Trans. Inf. Theory*, 54(5):2354–2357, 2008.
- [BCC⁺20] L. Budaghyan, M. Calderini, C. Carlet, R. S. Coulter, and I. Villa. Constructing APN functions through isotopic shifts. *IEEE Trans. Inf. Theory*, 66(8):5299–5309, 2020.
- [BCC⁺21] L. Budaghyan, M. Calderini, C. Carlet, R. S. Coulter, and I. Villa. Generalized isotopic shift construction for APN functions. *Des. Codes Cryptogr.*, 89(1):19–32, 2021.
- [BCL08] L. Budaghyan, C. Carlet, and G. Leander. Two classes of quadratic APN binomials inequivalent to power functions. *IEEE Trans. Inf. Theory*, 54(9):4218–4229, 2008.
- [BCL09a] L. Budaghyan, C. Carlet, and G. Leander. Constructing new APN functions from known ones. *Finite Fields Their Appl.*, 15(2):150–159, 2009.
- [BCL09b] L. Budaghyan, C. Carlet, and G. Leander. On a construction of quadratic apn functions. In *2009 IEEE Information Theory Workshop*, pages 374–378. IEEE, 2009.
- [BCP97] W. Bosma, J. Cannon, and C. Playoust. The Magma algebra system. I. The user language. *J. Symbolic Comput.*, 24(3-4):235–265, 1997.
- [BCP06] L. Budaghyan, C. Carlet, and A. Pott. New classes of almost bent and almost perfect nonlinear polynomials. *IEEE Trans. Information Theory*, 52(3):1141–1152, 2006.
- [BCV20] L. Budaghyan, M. Calderini, and I. Villa. On equivalence between known families of quadratic APN functions. *Finite Fields Their Appl.*, 66:101704, 2020.
- [BDMW10] K. A. Browning, J. F. Dillon, M. T. McQuistan, and A. J. Wolfe. An APN permutation in dimension six. *Finite Fields: theory and applications*, 518:33–42, 2010.
- [BHK20] L. Budaghyan, T. Helleseht, and N. Kaleyski. A new family of APN quadrinomials. *IEEE Trans. Inf. Theory*, 66(11):7081–7087, 2020.
- [BL08] M. Brinkmann and G. Leander. On the classification of APN functions up to dimension five. *Des. Codes Cryptogr.*, 49(1-3):273–288, 2008.
- [BL21] C. Beierle and G. Leander. New instances of quadratic apn functions in small dimension. Dataset, Version 2.1, 2021. DOI: 10.5281/zenodo.4738942.
- [BS91] E. Biham and A. Shamir. Differential cryptanalysis of des-like cryptosystems. *J. Cryptology*, 4(1):3–72, 1991.
- [Car18] C. Carlet. Characterizations of the differential uniformity of vectorial functions by the Walsh transform. *IEEE Trans. Inf. Theory*, 64(9):6443–6453, 2018.
- [Car20] C. Carlet. Bounds on the nonlinearity of differentially uniform functions by means of their image set size, and on their distance to affine functions. Cryptology ePrint Archive, Report 2020/1529, 2020. <https://eprint.iacr.org/2020/1529>.
- [Car21] C. Carlet. *Boolean Functions for Cryptography and Coding Theory*. Cambridge University Press, 2021.

- [cbe] cbe90. cbe90/quadratic_apn: Tree Search for Quadratic APN Functions v1.0. Software, Zenodo, 2020. DOI: 10.5281/zenodo.4305864.
- [CCP21] A. Canteaut, A. Couvreur, and L. Perrin. Recovering or testing extended-affine equivalence. *CoRR*, abs/2103.00078, 2021.
- [CCZ98] C. Carlet, P. Charpin, and V. Zinoviev. Codes, bent functions and permutations suitable for des-like cryptosystems. *Des. Codes Cryptogr.*, 15(2):125–156, 1998.
- [CHP17] C. Carlet, A. Heuser, and S. Picek. Trade-offs for s-boxes: Cryptographic properties and side-channel resilience. In Dieter Gollmann, Atsuko Miyaji, and Hiroaki Kikuchi, editors, *Applied Cryptography and Network Security - 15th International Conference, ACNS 2017, Kanazawa, Japan, July 10-12, 2017, Proceedings*, volume 10355 of *Lecture Notes in Computer Science*, pages 393–414. Springer, 2017.
- [CP20] A. Canteaut and L. Perrin. How to take a function apart with SboxU. *The 5th International Workshop on Boolean Functions and their Applications (BFA), invited talk*, 2020.
- [DF04] D. S. Dummit and R. M. Foote. *Abstract algebra*. John Wiley and Sons, Inc., 2004.
- [EP09] Y. Edel and A. Pott. A new almost perfect nonlinear function which is not quadratic. *Adv. Math. Commun.*, 3(1):59–81, 2009.
- [KI20a] K. Kalgin and V. Idrisova. The classification of quadratic apn functions in 7 variables. Cryptology ePrint Archive, Report 2020/1515, 2020. <https://eprint.iacr.org/2020/1515>.
- [KI20b] K. Kalgin and V. Idrisova. On secondary and cyclic approaches to search for quadratic apn functions. *The 11th SEquences and Their Applications (SETA)*, 2020.
- [Lan12] P. Langevin. Classification of APN cubics in dimension 6 over GF(2). <http://langevin.univ-tln.fr/project/apn-6/apn-6.html>, 2012. accessed October 6, 2021.
- [Mat93] M. Matsui. Linear cryptanalysis method for DES cipher. In Tor Helleseth, editor, *Advances in Cryptology - EUROCRYPT '93, Workshop on the Theory and Application of Cryptographic Techniques, Lofthus, Norway, May 23-27, 1993, Proceedings*, volume 765 of *Lecture Notes in Computer Science*, pages 386–397. Springer, 1993.
- [NK92] K. Nyberg and L. R. Knudsen. Provable security against differential cryptanalysis. In Ernest F. Brickell, editor, *Advances in Cryptology - CRYPTO '92, 12th Annual International Cryptology Conference, Santa Barbara, California, USA, August 16-20, 1992, Proceedings*, volume 740 of *Lecture Notes in Computer Science*, pages 566–574. Springer, 1992.
- [Per17] L. Perrin. sboxU. *GitHub repository*, 2017. Available via <https://github.com/lpp-crypto/sboxU>.

- [SZZ94] J. Seberry, X. M. Zhang, and Y. Zheng. Relationships among nonlinear criteria (extended abstract). In Alfredo De Santis, editor, *Advances in Cryptology - EUROCRYPT '94, Workshop on the Theory and Application of Cryptographic Techniques, Perugia, Italy, May 9-12, 1994, Proceedings*, volume 950 of *Lecture Notes in Computer Science*, pages 376–388. Springer, 1994.
- [Tan19] H. Taniguchi. On some quadratic APN functions. *Des. Codes Cryptogr.*, 87(9):1973–1983, 2019.
- [WTG13] G. Weng, Y. Tan, and G. Gong. On quadratic almost perfect nonlinear functions and their related algebraic object. *International Workshop on Coding and Cryptography (WCC)*, 2013.
- [YKBL20] Y. Yu, N. Kaleyski, L. Budaghyan, and Y. Li. Classification of quadratic apn functions with coefficients in F_2 for dimensions up to 9. *Finite Fields and Their Applications*, 68:101733, 2020.
- [Yos12] S. Yoshiara. Equivalences of quadratic apn functions. *J. Algebr. Comb.*, 35(3):461–475, 2012.
- [Yos16] S. Yoshiara. Equivalences of power apn functions with power or quadratic apn functions. *J. Algebr. Comb.*, 44(3):561–585, 2016.
- [YP21] Y. Yu and L. Perrin. Constructing more quadratic apn functions with the qam method. *The 6th International Workshop on Boolean Functions and their Applications (BFA)*, 2021.
- [YWL13] Y. Yu, M. Wang, and Y. Li. A matrix approach for constructing quadratic APN functions. Cryptology ePrint Archive, Report 2013/007, 2013. <https://eprint.iacr.org/2013/007>.
- [YWL14] Y. Yu, M. Wang, and Y. Li. A matrix approach for constructing quadratic APN functions. *Des. Codes Cryptogr.*, 73(2):587–600, 2014.