Constructing More Quadratic APN Functions with the QAM Method

Yuyin Yu[1] Léo Perrin[2]

[1]Guangzhou University, Guangzhou, China
[2]Inria, France

September 6th, 2021
BFA 2021
Rosendal, Norway

Outline

Plan of this Section

Why Generate Quadratic APN Functions?

Definition (APN function)

A function $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$ is Almost Perfect Non-linear (APN) if

$$F(x + a) - F(x) = b$$

has at most two solutions for all $a \neq 0, b$.

Why Generate Quadratic APN Functions?

Definition (APN function)

A function $F : \mathbb{F}_2^n \rightarrow \mathbb{F}_2^m$ is Almost Perfect Non-linear (APN) if

$$F(x + a) - F(x) = b$$

has at most two solutions for all $a \neq 0, b$.

The Big APN Problem

Does there exist an APN permutation on $\mathbb{F}_2^n$ for n even?

Why Generate Quadratic APN Functions?

Definition (APN function)

A function $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$ is Almost Perfect Non-linear (APN) if

$$F(x + a) - F(x) = b$$

has at most two solutions for all $a \neq 0, b$.

The Big APN Problem

Does there exist an APN permutation on $\mathbb{F}_2^n$ for n even?

- $n = 4$
- $n = 6$
- $n \geq 8$

Why Generate Quadratic APN Functions?

Definition (APN function)

A function $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$ is Almost Perfect Non-linear (APN) if

$$F(x + a) - F(x) = b$$

has at most two solutions for all $a \neq 0, b$.

The Big APN Problem

Does there exist an APN permutation on $\mathbb{F}_2^n$ for n even?

- $n = 4$ **No.**
- $n = 6$ **Yes! [Dillon et al. 09]**
- $n \geq 8$ **???**

Why Generate Quadratic APN Functions?

Definition (APN function)

A function $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$ is Almost Perfect Non-linear (APN) if

$$F(x + a) - F(x) = b$$

has at most two solutions for all $a \neq 0$, b.

The Big APN Problem

Does there exist an APN permutation on $\mathbb{F}_2^n$ for n even?

- $n = 4$ **No.**
- $n = 6$ **Yes! [Dillon et al. 09]**
  Find permutation in the *CCZ-class* of a known APN function (the "Kim mapping")
- $n \geq 8$ **???**

Equivalence Relations

Definition (Affine-Equivalence)

F and G are *affine equivalent* if $G(x) = (B \circ F \circ A)(x)$, where A, B are affine permutations.

Equivalence Relations

Definition (Affine-Equivalence)

F and G are *affine equivalent* if $G(x) = (B \circ F \circ A)(x)$, where A, B are affine permutations.

Definition (EA-Equivalence)

F and G are *E(xtented) A(ffine) equivalent* if $G(x) = (B \circ F \circ A)(x) + C(x)$, where A, B, C are affine and A, B are permutations.

Definition (CCZ-Equivalence)

$F : \mathbb{F}_2^n \rightarrow \mathbb{F}_2^m$ and $G : \mathbb{F}_2^n \rightarrow \mathbb{F}_2^m$ are *C(arlet)-C(harpin)-Z(inoviev) equivalent* if

$$\Gamma_G = \left\{(x, G(x)), \forall x \in \mathbb{F}_2^n\right\} = L\left(\left\{(x, F(x)), \forall x \in \mathbb{F}_2^n\right\}\right) = L(\Gamma_F),$$

where $L : \mathbb{F}_2^{n+m} \rightarrow \mathbb{F}_2^{n+m}$ is an affine permutation.

Already Known 8-bit Quadratic APN permutations

1. “Switching” method [Edel Pott 2009]

Already Known 8-bit Quadratic APN permutations

1. “Switching” method [Edel Pott 2009]
2. QAM method [Yu et al. 2014]
   (see rest of this talk).

Already Known 8-bit Quadratic APN permutations

1. "Switching" method [Edel Pott 2009]
2. QAM method [Yu et al. 2014]
   (see rest of this talk).
3. Self-equivalent functions [Beierle Leander 2020]:
   Look for functions $F : \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ such that

$$F \circ A = B \circ F$$

   for linear permutations A and B.

Already Known 8-bit Quadratic APN permutations

1. "Switching" method [Edel Pott 2009]
2. QAM method [Yu et al. 2014]
   (see rest of this talk).
3. Self-equivalent functions [Beierle Leander 2020]:
   Look for functions $F : \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ such that

   $$F \circ A = B \circ F$$

   for linear permutations A and B.
4. QIC [Ghosh Perrin 2020]
   (see next talk).

Already Known 8-bit Quadratic APN permutations

1. "Switching" method [Edel Pott 2009] **23**
2. QAM method [Yu et al. 2014] **8179**
   (see rest of this talk).
3. Self-equivalent functions [Beierle Leander 2020]: **12812+188**
   Look for functions $F : \mathbb{F}_2^n \to \mathbb{F}_2^n$ such that

   $$F \circ A = B \circ F$$

   for linear permutations A and B.
4. QIC [Ghosh Perrin 2020] **–**
   (see next talk).

Already Known 8-bit Quadratic APN permutations

1. "Switching" method [Edel Pott 2009] **23**
2. QAM method [Yu et al. 2014] **8179**
   (see rest of this talk).
3. Self-equivalent functions [Beierle Leander 2020]: **12812+188**
   Look for functions $F : \mathbb{F}_2^n \to \mathbb{F}_2^n$ such that

$$F \circ A = B \circ F$$

   for linear permutations A and B.
4. QIC [Ghosh Perrin 2020] **–**
   (see next talk).

Total (without redundancy) **21112**

Some Questions

1. How many quadratic APN functions exist in dimension 8?

Some Questions

1. How many quadratic APN functions exist in dimension 8?

2. What’s the overlap between the classes of known funcitons?

Some Questions

1. How many quadratic APN functions exist in dimension 8?

   Conjecture: $> 50,000$

2. What's the overlap between the classes of known funcitons?

Some Questions

1. How many quadratic APN functions exist in dimension 8?

   Conjecture: $> 50,000$

2. What's the overlap between the classes of known funcitons?

   not much!

Plan of this Section

Definition of the QAM

Definition (Quadratic Homogeneous Functions)

Quadratic functions without linear or constant terms are called **quadratic homogeneous functions**:

$$F(x) = \sum_{1 \leq j < i \leq n} c_{i,j} x^{2^{i-1}+2^{j-1}} \in \mathbb{F}_{2^n}[x].$$

Definition (QAM)

Let $H = (h_{i,j})_{n \times n}$ be an $n \times n$ matrix of $\mathbb{F}_{2^n}$. It is a **Quadratic APN Matrix (QAM)** if

1. it is symmetric and the elements in its main diagonal are all zeros; and
2. every nonzero linear combination of its rows has rank $n - 1$.

Properties

$$H = \begin{pmatrix} 0 & g^{34} & g^{81} & g^{83} & g^{170} & g^{106} & \mathbf{x}_{13} & \mathbf{x}_{7} \\ g^{34} & 0 & g^{68} & g^{162} & g^{166} & g^{85} & \mathbf{x}_{12} & \mathbf{x}_{6} \\ g^{81} & g^{68} & 0 & g^{136} & g^{69} & g^{77} & \mathbf{x}_{11} & \mathbf{x}_{5} \\ g^{83} & g^{162} & g^{136} & 0 & g^{17} & g^{138} & \mathbf{x}_{10} & \mathbf{x}_{4} \\ g^{170} & g^{166} & g^{69} & g^{17} & 0 & g^{34} & \mathbf{x}_{9} & \mathbf{x}_{3} \\ g^{106} & g^{85} & g^{77} & g^{138} & g^{34} & 0 & \mathbf{x}_{8} & \mathbf{x}_{2} \\ \mathbf{x}_{13} & \mathbf{x}_{12} & \mathbf{x}_{11} & \mathbf{x}_{10} & \mathbf{x}_{9} & \mathbf{x}_{8} & 0 & \mathbf{x}_{1} \\ \mathbf{x}_{7} & \mathbf{x}_{6} & \mathbf{x}_{5} & \mathbf{x}_{4} & \mathbf{x}_{3} & \mathbf{x}_{2} & \mathbf{x}_{1} & 0 \end{pmatrix}$$

Theorem (Yu et al.[1])

There exists a one to one correspondence between quadratic homogeneous APN functions and QAMs.

[1] Y. Yu, M. Wang, Y. Li, A matrix approach for constructing quadratic APN functions. Designs Codes and Cryptography 73, p.587-600 (2014).

Generating New Functions

1. Take an APN function $(x \mapsto x^3)$

Generating New Functions

1. Take an APN function $(x \mapsto x^3)$
2. Compute its QAM.

Generating New Functions

1. Take an APN function $(x \mapsto x^3)$
2. Compute its QAM.
3. Let the last two rows/columns be variables $\{x_1, \dots\}$:

$$H = \begin{pmatrix} 0 & g^{34} & g^{81} & g^{83} & g^{170} & g^{106} & \mathbf{x_{13}} & \mathbf{x_7} \\ g^{34} & 0 & g^{68} & g^{162} & g^{166} & g^{85} & \mathbf{x_{12}} & \mathbf{x_6} \\ g^{81} & g^{68} & 0 & g^{136} & g^{69} & g^{77} & \mathbf{x_{11}} & \mathbf{x_5} \\ g^{83} & g^{162} & g^{136} & 0 & g^{17} & g^{138} & \mathbf{x_{10}} & \mathbf{x_4} \\ g^{170} & g^{166} & g^{69} & g^{17} & 0 & g^{34} & \mathbf{x_9} & \mathbf{x_3} \\ g^{106} & g^{85} & g^{77} & g^{138} & g^{34} & 0 & \mathbf{x_8} & \mathbf{x_2} \\ \mathbf{x_{13}} & \mathbf{x_{12}} & \mathbf{x_{11}} & \mathbf{x_{10}} & \mathbf{x_9} & \mathbf{x_8} & 0 & \mathbf{x_1} \\ \mathbf{x_7} & \mathbf{x_6} & \mathbf{x_5} & \mathbf{x_4} & \mathbf{x_3} & \mathbf{x_2} & \mathbf{x_1} & 0 \end{pmatrix}$$

Generating New Functions

1. Take an APN function $(x \mapsto x^3)$
2. Compute its QAM.
3. Let the last two rows/columns be variables $\{x_1, \dots\}$:

$$H = \begin{pmatrix} 0 & g^{34} & g^{81} & g^{83} & g^{170} & g^{106} & \mathbf{x_{13}} & \mathbf{x_7} \\ g^{34} & 0 & g^{68} & g^{162} & g^{166} & g^{85} & \mathbf{x_{12}} & \mathbf{x_6} \\ g^{81} & g^{68} & 0 & g^{136} & g^{69} & g^{77} & \mathbf{x_{11}} & \mathbf{x_5} \\ g^{83} & g^{162} & g^{136} & 0 & g^{17} & g^{138} & \mathbf{x_{10}} & \mathbf{x_4} \\ g^{170} & g^{166} & g^{69} & g^{17} & 0 & g^{34} & \mathbf{x_9} & \mathbf{x_3} \\ g^{106} & g^{85} & g^{77} & g^{138} & g^{34} & 0 & \mathbf{x_8} & \mathbf{x_2} \\ \mathbf{x_{13}} & \mathbf{x_{12}} & \mathbf{x_{11}} & \mathbf{x_{10}} & \mathbf{x_9} & \mathbf{x_8} & 0 & \mathbf{x_1} \\ \mathbf{x_7} & \mathbf{x_6} & \mathbf{x_5} & \mathbf{x_4} & \mathbf{x_3} & \mathbf{x_2} & \mathbf{x_1} & 0 \end{pmatrix}$$

4. Let $\{x_1, \dots\}$ take different values and check if we have a QAM.

Sorting the Result

This approach works (see later)!

Sorting the Result

This approach works (see later)!

One Problem

How to ensure that we do not generate the same function multiple times?

Sorting the Result

This approach works (see later)!

One Problem

How to ensure that we do not generate the same function multiple times?

A better statement

How to partition the functions obtained into

Sorting the Result

This approach works (see later)!

One Problem

How to ensure that we do not generate the same function multiple times?

A better statement

How to partition the functions obtained into CCZ-equivalence classes?

Invariant-based Approach

Theorem ([Yos12][2])

Quadratic APN functions are CCZ-equivalent if and only if they are EA-equivalent.

[2] Satoshi Yoshiara. Equivalences of quadratic apn functions. Journal of Algebraic Combinatorics, 35(3):461–475, 2012.

Invariant-based Approach

Theorem ([Yos12][2])

Quadratic APN functions are CCZ-equivalent if and only if they are EA-equivalent.

We use EA-class invariants:

[2] Satoshi Yoshiara. Equivalences of quadratic apn functions. Journal of Algebraic Combinatorics, 35(3):461–475, 2012.

Invariant-based Approach

Theorem ([Yos12][2])

Quadratic APN functions are CCZ-equivalent if and only if they are EA-equivalent.

We use EA-class invariants:

δ-rank; Γ-rank: the ranks of $2^{2n} \times 2^{2n}$ matrices computed from F.

[2] Satoshi Yoshiara. Equivalences of quadratic apn functions. Journal of Algebraic Combinatorics, 35(3):461–475, 2012.

Invariant-based Approach

Theorem ([Yos12][2])

Quadratic APN functions are CCZ-equivalent if and only if they are EA-equivalent.

We use EA-class invariants:

δ-rank; Γ-rank: the ranks of $2^{2n} \times 2^{2n}$ matrices computed from F.

Thickness spectrum: A property of the *Walsh zeroes* of F.

$\Sigma_F^k(0)$: How many tuples $(x_1, \ldots, x_k)$ such that:

$$x_1 + \ldots + x_k = 0; \text{ and } F(x_1) + \ldots + F(x_k) = 0 .$$

[2] Satoshi Yoshiara. Equivalences of quadratic apn functions. Journal of Algebraic Combinatorics, 35(3):461–475, 2012.

Invariant-based Approach

Theorem ([Yos12][2])

Quadratic APN functions are CCZ-equivalent if and only if they are EA-equivalent.

We use EA-class invariants:

δ-rank; Γ-rank: the ranks of $2^{2n} \times 2^{2n}$ matrices computed from F.

Thickness spectrum: A property of the *Walsh zeroes* of F.

$\Sigma_F^k(0)$: How many tuples $(x_1, \ldots, x_k)$ such that:

$$x_1 + \ldots + x_k = 0; \text{ and } F(x_1) + \ldots + F(x_k) = 0 .$$

Ortho-derivative: $\pi_F : \mathbb{F}_2^n \to \mathbb{F}_2^n$ is the unique function such that $\pi_F(0) = 0$ and, for all x, a:

$$\pi_F(a) \cdot \big(F(x+a) + F(x) + F(a) + F(0)\big) = 0.$$

Its affine equivalence-class is an EA-class invariant.

[2]Satoshi Yoshiara. Equivalences of quadratic apn functions. Journal of Algebraic Combinatorics, 35(3):461–475, 2012.

Implementation aspects

$$F : \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$$
$$\mathtt{F} = [F(0), F(1), \ldots, F(2^n - 1)]$$

Implementation aspects

$$F : \mathbb{F}_2^n \to \mathbb{F}_2^n$$
$$\texttt{F} = [F(0), F(1), \ldots, F(2^n - 1)]$$

Name	Complexity	sboxU function
δ-ranks	$O(2^{2\omega n})$	`delta_rank(F)`
Γ-ranks	$O(2^{2\omega n})$	`gamma_rank(F)`
Thickness spectrum	?	`thickness_spectrum(F)`
Σ_F^k	$O(n2^{2n})$	`sigma_multiplicities(F, k)`
π_F	$O(2^{2n})$	`ortho_derivative_label(F)`

Implementation aspects

$$F : \mathbb{F}_2^n \to \mathbb{F}_2^n$$
$$\texttt{F} = [F(0), F(1), \ldots, F(2^n - 1)]$$

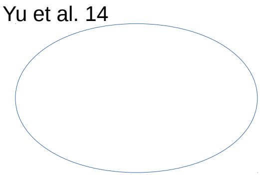
Name	Complexity	sboxU function
δ-ranks	$O(2^{2\omega n})$	`delta_rank(F)`
Γ-ranks	$O(2^{2\omega n})$	`gamma_rank(F)`
Thickness spectrum	?	`thickness_spectrum(F)`
Σ_F^k	$O(n2^{2n})$	`sigma_multiplicities(F, k)`
π_F	$O(2^{2n})$	`ortho_derivative_label(F)`

Are my APN functions new?

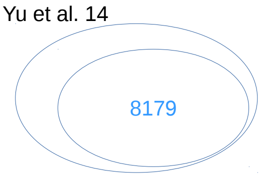
```
from collections import defaultdict
from sboxU import *

ea_counters = defaultdict

known_apn_functions = eightBitAPN.all_quadratics()
for f in known_apn_functions:
    ea_counters[ortho_derivative_label(f)] += 1

new_QAMs = [[0, ..., 255], ... ]
updated_apn_functions = known_apn_functions[:]
for f in new_QAMs:
    l = ortho_derivative_label(f)
    ea_counters[l] += 1
    if ea_counters[l] == 1:
        updated_apn_functions.append(f)
```

Plan of this Section

8-bit Quadratic APN Generation



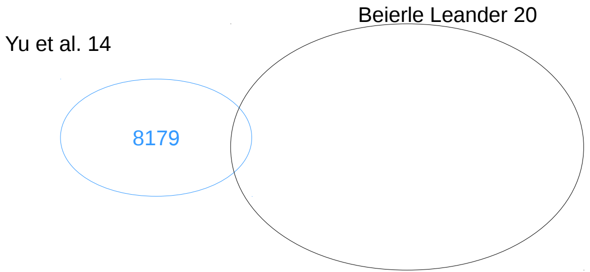
8-bit Quadratic APN Generation



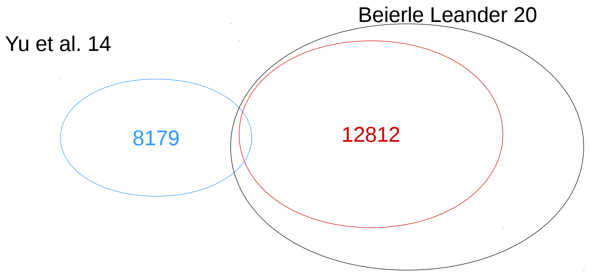
8-bit Quadratic APN Generation

Yu et al. 14

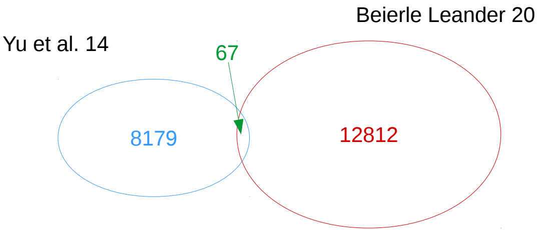
8-bit Quadratic APN Generation



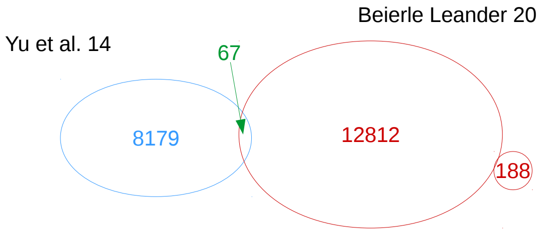
8-bit Quadratic APN Generation



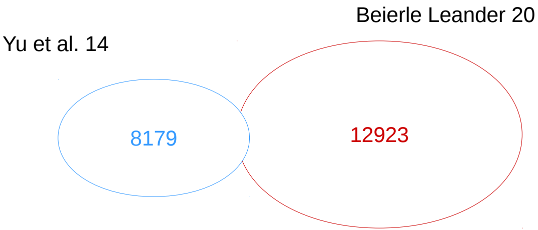
8-bit Quadratic APN Generation



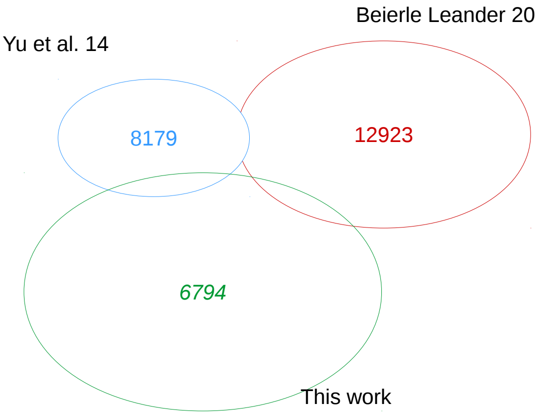
8-bit Quadratic APN Generation



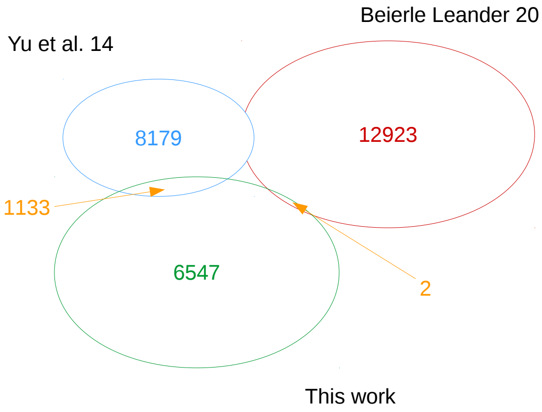
8-bit Quadratic APN Generation



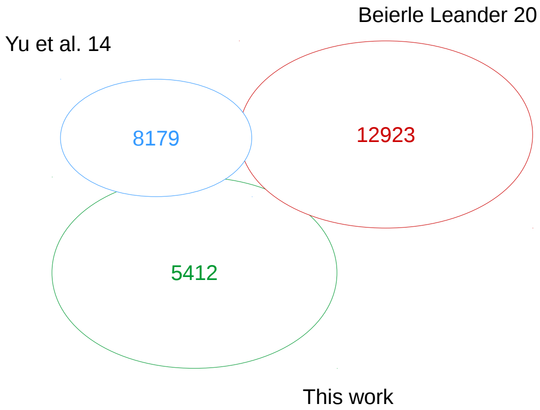
8-bit Quadratic APN Generation



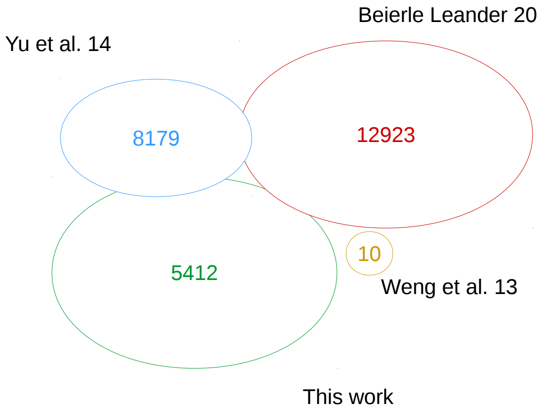
8-bit Quadratic APN Generation



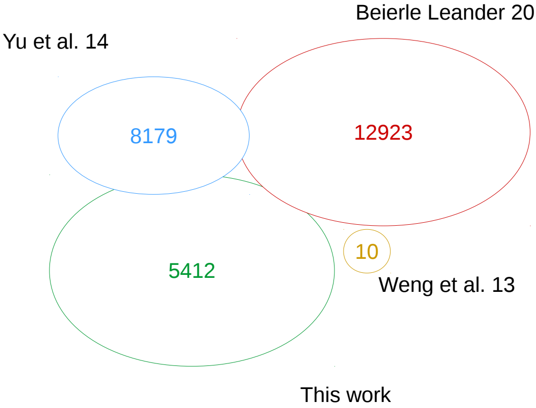
8-bit Quadratic APN Generation



8-bit Quadratic APN Generation



8-bit Quadratic APN Generation




```
len(sboxU.eightBitAPN.all_quadratics()) = 26524
```

Total Number of APN Functions

A simple test

Knowing that k quadratic APN functions of $\mathbb{F}_2^n$ have been generated using QAMs, what is the probability P_k^n that the next generated function is new?

Total Number of APN Functions

A simple test

Knowing that k quadratic APN functions of $\mathbb{F}_2^n$ have been generated using QAMs, what is the probability P_k^n that the next generated function is new?

$n = 6$ For $k = 6$ (out of 13), $P_6 \approx 75\%$

Total Number of APN Functions

A simple test

Knowing that k quadratic APN functions of $\mathbb{F}_2^n$ have been generated using QAMs, what is the probability P_k^n that the next generated function is new?

- $n = 6$ For $k = 6$ (out of 13), $P_6 \approx 75\%$
- $n = 7$ For $k = 230$ (out of 488), $P_{230} \approx 79\%$

Total Number of APN Functions

A simple test

Knowing that k quadratic APN functions of $\mathbb{F}_2^n$ have been generated using QAMs, what is the probability P_k^n that the next generated function is new?

- $n = 6$ For $k = 6$ (out of 13), $P_6 \approx 75\%$
- $n = 7$ For $k = 230$ (out of 488), $P_{230} \approx 79\%$
- $n = 8$ For $k = 25624$, $P_{25624} \approx 79\%$

Total Number of APN Functions

A simple test

Knowing that k quadratic APN functions of $\mathbb{F}_2^n$ have been generated using QAMs, what is the probability P_k^n that the next generated function is new?

- $n = 6$ For $k = 6$ (out of 13), $P_6 \approx 75\%$
- $n = 7$ For $k = 230$ (out of 488), $P_{230} \approx 79\%$
- $n = 8$ For $k = 25624$, $P_{25624} \approx 79\%$

Conjecture

There are at least $50,000$ quadratic APN functions on 8 bits.

How to get them?

Using the QAM method

For a given n, how many QAMs do we need to generate to obtain all ℓ_n quadratic APN functions?

How to get them?

Using the QAM method

For a given n, how many QAMs do we need to generate to obtain all ℓ_n quadratic APN functions?

$n = 6$ We need $200 \approx 16 \times \ell_6$ QAMs to obtain all $\ell_n = 13$

How to get them?

Using the QAM method

For a given n, how many QAMs do we need to generate to obtain all ℓ_n quadratic APN functions?

$n = 6$ We need $200 \approx 16 \times \ell_6$ QAMs to obtain all $\ell_n = 13$

$n = 7$ We need $3000 \approx 8 \times \ell_7$ QAMs to obtain all $\ell_n = 488$

How to get them?

Using the QAM method

For a given n, how many QAMs do we need to generate to obtain all ℓ_n quadratic APN functions?

$n = 6$ We need $200 \approx 16 \times \ell_6$ QAMs to obtain all $\ell_n = 13$

$n = 7$ We need $3000 \approx 8 \times \ell_7$ QAMs to obtain all $\ell_n = 488$

$n = 8$?

How to get them?

Using the QAM method

For a given n, how many QAMs do we need to generate to obtain all ℓ_n quadratic APN functions?

- $n = 6$ We need $200 \approx 16 \times \ell_6$ QAMs to obtain all $\ell_n = 13$
- $n = 7$ We need $3000 \approx 8 \times \ell_7$ QAMs to obtain all $\ell_n = 488$
- $n = 8$?

Conjecture

For $n = 8$, we would need to generate $4 \times \ell_8 \approx 200,000$ QAMs to generate all of them, i.e. about 50 CPU·year.

Plan of this Section

Conclusion

There are many 8-bit quadratic APN functions!