How to Take a Function Apart with SboxU
(Also Featuring some New Results on Ortho-Derivatives)

Anne Canteaut\textsuperscript{1}, Léo Perrin\textsuperscript{1}

\textsuperscript{1}Inria, France

leo.perrin@inria.fr

@lpp_crypto

Boolean Functions and their Applications 2020
A wild vectorial Boolean function appears!

What do you do?
A wild vectorial Boolean function appears!
A wild vectorial Boolean function appears!

What do you do?
Outline

1. Basic Functionalities
2. CCZ-Equivalence
3. Ortho-Derivative
4. Conclusion
Plan of this Section

1. Basic Functionalities
   - Installation
   - Core Functionalities

2. CCZ-Equivalence

3. Ortho-Derivative

4. Conclusion
Plan of this Section

1. Basic Functionalities
   - Installation
   - Core Functionalities

2. CCZ-Equivalence

3. Ortho-Derivative

4. Conclusion
How to

- You need to have SAGE installed
- Then head to https://github.com/lpp-crypto/sboxU
Sbox from SAGE vs. sboxU

There are already many functions for investigating vectorial boolean functions in SAGE:

- Class SBox from `sage.crypto.sbox` (or `sage.crypto.mq.sbox` in older versions)
- Module `boolean_function` from `sage.crypto`
Sbox from SAGE vs. sboxU

There are already many functions for investigating vectorial boolean functions in SAGE:

- Class SBox from sage.crypto.sbox (or sage.crypto.mq.sbox in older versions)
- Module boolean_function from sage.crypto

### SAGE SBox

- Supports output size \( \neq \) input size
- Sub-routines written in Python or Cython
- Built-in SAGE

### sboxU

- **Assumes** output size = input size
- Sub-routines written in Python or multi-threaded C++
- Cutting functionalities
Plan of this Section

1. Basic Functionalities
   - Installation
   - Core Functionalities

2. CCZ-Equivalence

3. Ortho-Derivative

4. Conclusion
Some Tools

1. DDT/LAT (+ Pollock representation thereof)
2. ANF, algebraic degree
3. Finite field arithmetic
4. Linear mappings

Demo
Plan of this Section

1. Basic Functionalities
2. CCZ-Equivalence
   - Definition and Basic Theorems
   - How Can sboxU Help?
3. Ortho-Derivative
4. Conclusion
Plan of this Section

1. Basic Functionalities
2. CCZ-Equivalence
   - Definition and Basic Theorems
   - How Can sboxU Help?
3. Ortho-Derivative
4. Conclusion
CCZ- and EA-equivalence

Definition (CCZ-Equivalence)

\[ F : \mathbb{F}_2^n \to \mathbb{F}_2^m \text{ and } G : \mathbb{F}_2^n \to \mathbb{F}_2^m \text{ are } \text{C(arlet)}{-}\text{C(harpin)}{-}\text{Z(inoviev)} \text{ equivalent if} \]

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

where \( L : \mathbb{F}_2^{n+m} \to \mathbb{F}_2^{n+m} \) is an affine permutation.
CCZ- and EA-equivalence

**Definition (CCZ-Equivalence)**

Let $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$ and $G : \mathbb{F}_2^n \to \mathbb{F}_2^m$ be functions. They are \textit{C(arlet)-C(harpin)-Z(inoviev) equivalent} if

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

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

**Definition (EA-Equivalence; EA-mapping)**

Let $F$ and $G$ be functions. $F$ and $G$ are \textit{E(xtended) 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; so that

$$
\{(x, G(x)), \forall x \in \mathbb{F}_2^n\} = \begin{bmatrix} A^{-1} & 0 \\ CA^{-1} & B \end{bmatrix} (\{(x, F(x)), \forall x \in \mathbb{F}_2^n\}).
$$
Some Algorithmic Problems with CCZ-Equivalence

CCZ-class

$F$
Some Algorithmic Problems with CCZ-Equivalence

CCZ-class

EA-class  EA-class  EA-class  EA-class  EA-class

$F$
Some Algorithmic Problems with CCZ-Equivalence

CCZ-class

<table>
<thead>
<tr>
<th>EA-class</th>
<th>EA-class</th>
<th>EA-class</th>
<th>EA-class</th>
<th>EA-class</th>
</tr>
</thead>
<tbody>
<tr>
<td>$F$</td>
<td>$F_1$</td>
<td>$F_2$</td>
<td>$F_3$</td>
<td>$F_4$</td>
</tr>
</tbody>
</table>
Some Algorithmic Problems with CCZ-Equivalence

CCZ-class

\[ F \quad F'_1 \quad F_2 \quad F_3 \quad F_4 \]

EA-class  EA-class  EA-class  EA-class  EA-class
Some Algorithmic Problems with CCZ-Equivalence

CCZ-class

F′
F
F1
F2
F3
F4
G
Plan of this Section

1. Basic Functionalities
2. CCZ-Equivalence
   - Definition and Basic Theorems
   - How Can sboxU Help?
3. Ortho-Derivative
4. Conclusion
Exploring a CCZ-class

Algorithms used here are based on:

- an efficient vector space search algorithm from “Anomalies and Vector Space Search: Tools for S-Box Analysis” (ASIACRYPT'19), and
- the framework based on Walsh zeroes we introduced in “On CCZ-equivalence, extended-affine equivalence, and function twisting”, FFA'19

Finding representatives of EA-classes

Demo
Exploring a CCZ-class

Algorithms used here are based on:

- an efficient vector space search algorithm from “Anomalies and Vector Space Search: Tools for S-Box Analysis” (ASIACRYPT'19), and

- the framework based on Walsh zeroes we introduced in “On CCZ-equivalence, extended-affine equivalence, and function twisting”, FFA'19

Finding representatives of EA-classes

Finding permutations!
Class Invariants

**Definition (Differential spectrum)**

Recall that $DDT_F[a, b] = \# \{ x, F(x + a) + F(x) = b \}$. The **differential spectrum** is the number of occurrences of each number in the DDT.
**Class Invariants**

**Definition (Differential spectrum)**

Recall that $\text{DDT}_F[a, b] = \# \left\{ x, F(x + a) + F(x) = b \right\}$. The **differential spectrum** is the number of occurrences of each number in the DDT.

**Definition (Walsh spectrum)**

Recall that $\mathcal{W}_F[a, b] = \sum_x (-1)^{a \cdot x + b \cdot F(x)}$. The **Walsh spectrum** is the number of occurrences of each number in the LAT. The **extended Walsh spectrum** considers only absolute values.
Definition (Differential spectrum)

Recall that $\text{DDT}_F[a, b] = \#\{x, F(x + a) + F(x) = b\}$. The **differential spectrum** is the number of occurrences of each number in the DDT.

Definition (Walsh spectrum)

Recall that $\mathcal{W}_F[a, b] = \sum_x (-1)^{a \cdot x + b \cdot F(x)}$. The **Walsh spectrum** is the number of occurrences of each number in the LAT. The **extended Walsh spectrum** considers only absolute values.

- Differential and extended Walsh spectra are constant in a **CCZ-class**.
- The algebraic degree and the **thickness spectrum** are constant in an **EA-class**.
Plan of this Section

1. Basic Functionalities
2. CCZ-Equivalence
3. Ortho-Derivative
   - Definition and Basic Theorems
   - Algorithmic Uses
   - Inverting the DDT of a Quadratic Function
4. Conclusion
Plan of this Section

1. Basic Functionalities
2. CCZ-Equivalence
3. Ortho-Derivative
   - Definition and Basic Theorems
   - Algorithmic Uses
   - Inverting the DDT of a Quadratic Function
4. Conclusion
Definition (Ortho-Derivative)

Let $F$ be a quadratic function of $\mathbb{F}_2^n$. The ortho-derivatives of $F$ are the functions of $\mathbb{F}_2^n$ such that

$$\forall x \in \mathbb{F}_2^n, \pi_F(a) \cdot \left( F(x + a) + F(x) + F(a) + F(0) \right) = 0.$$
Definition (Ortho-Derivative)

Let $F$ be a quadratic function of $\mathbb{F}_2^n$. The ortho-derivatives of $F$ are the functions of $\mathbb{F}_2^n$ such that

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

- $\pi_F(a)$ is orthogonal to the linear part of the hyperplane $\text{Im}(\Delta_{a}F)$
- $\pi_F$ can take any value in $0$. 
Lemma (Ortho-derivatives of APN functions)

$F$ is APN if and only if $\pi_F(a)$ is uniquely defined for all $a \in (\mathbb{F}_2^n)^*$. 

---

1See also A note on the properties of associated Boolean functions of quadratic APN functions by Anastasiya Gorodilova on ArXiv.
Lemma (Ortho-derivatives of APN functions)

\( F \) is APN if and only if \( \pi_F(a) \) is uniquely defined for all \( a \in (\mathbb{F}_2^n)^* \).

Lemma (Interaction with EA-equivalence)

If \( G = B \circ F \circ A + C \) where \( A \) and \( B \) are linear permutations and \( C \) is a linear function, then

\[ \pi_G = (B^T)^{-1} \circ \pi_F \circ A \]

\(^1\)See also A note on the properties of associated Boolean functions of quadratic APN functions by Anastasiya Gorodilova on ArXiv.
Basic Properties

Lemma (Ortho-derivatives of APN functions)

$F$ is APN if and only if $\pi_F(a)$ is uniquely defined for all $a \in (\mathbb{F}_2^n)^*$. 

Lemma (Interaction with EA-equivalence)

If $G = B \circ F \circ A + C$ where $A$ and $B$ are linear permutations and $C$ is a linear function, then

$$\pi_G = (B^T)^{-1} \circ \pi_F \circ A$$

It seems like\(^1\) the algebraic degree of the ortho-derivative of an APN function is always $n - 2$.

\(^1\)See also A note on the properties of associated Boolean functions of quadratic APN functions by Anastasiya Gorodilova on ArXiv.
Preimages of the Ortho-Derivative

Theorem (Linear Structures (APN case))

If

\[ T_F(b) = \{ x \in \mathbb{F}_2^n : \pi_F(x) = b \}, \]

then \( T_F(b) = \text{LS}(x \mapsto b \cdot F(x)) \).

Corollary

For any \( b \), \( T_F(b) \) is a linear subspace of \( \mathbb{F}_2^n \) whose dimension has the same parity as \( n \). Furthermore,

\[
(W_F[a, b])^2 \in \left\{ 0, 2^{n+\dim T_F(b)} \right\}
\]
Plan of this Section

1. Basic Functionalities
2. CCZ-Equivalence
3. Ortho-Derivative
   - Definition and Basic Theorems
   - Algorithmic Uses
   - Inverting the DDT of a Quadratic Function
4. Conclusion
Identifying EA- and CCZ-classes

Corollary (Ortho-derivatives of APN functions)

The differential and extended Walsh spectra of the ortho-derivative of an APN function is the same within an EA-class.
Identifying EA- and CCZ-classes

**Corollary (Ortho-derivatives of APN functions)**

The *differential* and *extended Walsh spectra* of the ortho-derivative of an APN function is the same within an EA-class.

**Observation**

In practice, these spectra differ from one EA-class to the next!
Identifying EA- and CCZ-classes

Corollary (Ortho-derivatives of APN functions)

The differential and extended Walsh spectra of the ortho-derivative of an APN function is the same within an EA-class.

Observation

In practice, these spectra differ from one EA-class to the next!

We can use this to very efficiently sort large numbers of quadratic functions into distinct EA-classes.
Plan of this Section

1. Basic Functionalities
2. CCZ-Equivalence
3. Ortho-Derivative
   - Definition and Basic Theorems
   - Algorithmic Uses
   - Inverting the DDT of a Quadratic Function
4. Conclusion
Is it possible to recover $F$ given $\pi_F$?
Is it possible to recover $F$ given $\pi_F$? Yes!

The Key Observation

We can write the scalar product $x \cdot y$ as $(\vec{x})^T \times \vec{y}$, where $\times$ is a matrix operation.
Is it possible to recover $F$ given $\pi_F$? Yes!

The Key Observation

We can write the scalar product $x \cdot y$ as $(\vec{x})^T \times \vec{y}$, where $\times$ is a matrix operation.

We represent $F$ as a vector of $\mathbb{F}_2^{n2^n}$ by concatenating the $n$-bit representation of each of the $2^n$ values $F(x)$:

$$\text{vec}(F) = \begin{bmatrix} F_0(0) \\ F_1(0) \\ \vdots \\ F_{n-1}(0) \\ F_0(1) \\ \vdots \\ F_{n-1}(2^n - 1) \end{bmatrix}.$$
Re-Defining Ortho-Derivatives

Let $G$ be a function and $\zeta_a(G)$ be a matrix defined by

1. $\zeta_G(a)[x, x] = G(a)^T$,  \hspace{1cm} $\zeta_G(a)[x, x + a] = G(a)^T$,  
2. $\zeta_G(a)[x, 0] = G(a)^T$,  \hspace{1cm} $\zeta_G(a)[x, a] = G(a)^T$,  

Re-Defining Ortho-Derivatives

Let $G$ be a function and $\zeta_a(G)$ be a matrix defined by

1. $\zeta_G(a)[x, x] = G(a)^T$, $\zeta_G(a)[x, x + a] = G(a)^T$,
2. $\zeta_G(a)[x, 0] = G(a)^T$, $\zeta_G(a)[x, a] = G(a)^T$,

so that

$$\zeta_G(a) \times \text{vec}(F) = \begin{bmatrix}
G(a) \cdot (F(0) + F(0 + a) + F(a) + F(0)) \\ G(a) \cdot (F(1) + F(1 + a) + F(a) + F(0)) \\
G(a) \cdot (F(2^n - 1) + F(2^n - 1 + a) + F(a) + F(0)) \\
\vdots
\end{bmatrix},$$
Re-Defining Ortho-Derivatives

Let $G$ be a function and $\zeta_a(G)$ be a matrix defined by

1. $\zeta_G(a)[x, x] = \overrightarrow{G(a)}^T$,  
   $\zeta_G(a)[x, x + a] = \overrightarrow{G(a)}^T$,  
2. $\zeta_G(a)[x, 0] = \overrightarrow{G(a)}^T$,  
   $\zeta_G(a)[x, a] = \overrightarrow{G(a)}^T$,

so that

$$
\zeta_G(a) \times \text{vec}(F) = \begin{bmatrix}
G(a) \cdot (F(0) + F(0 + a) + F(a) + F(0)) \\
G(a) \cdot (F(1) + F(1 + a) + F(a) + F(0)) \\
\vdots \\
G(a) \cdot (F(2^n - 1) + F(2^n - 1 + a) + F(a) + F(0))
\end{bmatrix},
$$

from which we deduce that if $\pi_F$ is an ortho-derivative of $F$ then

$$
\text{vec}(F) \in \ker (\zeta(\pi_F)) \quad \text{where} \quad \zeta(\pi_F) = \begin{bmatrix}
\zeta_0(\pi_F) \\
\vdots \\
\zeta_{2^n-1}(\pi_F)
\end{bmatrix}.
$$
Inverting the DDT of a Quadratic Function

1. Find a DDT,
2. deduce the corresponding $\pi$,
3. build $\zeta(\pi)$,
4. find ker($\zeta(\pi)$),
5. obtain vec($F$)!

2 Tricks are used to get rid of redundancies in $\zeta$, and trivial solutions.
Inverting the DDT of a Quadratic Function

1. Find a DDT,
2. deduce the corresponding $\pi$,
3. build $\zeta(\pi)$,
4. find ker $(\zeta(\pi))$,
5. obtain vec($F$)!

In practice, starting from “cleverly” built functions $\pi$ yields $\zeta(\pi)$ with empty² kernels...

²Tricks are used to get rid of redundancies in $\zeta$, and trivial solutions.
Plan of this Section

1. Basic Functionalities
2. CCZ-Equivalence
3. Ortho-Derivative
4. Conclusion
Conclusion

Go an use sboxU!  https://github.com/lpp-crypto/sboxU

Send me an email (leo.perrin@inria.fr) if you want to join the sboxU mailing list.
Conclusion

Go an use sboxU! https://github.com/lpp-crypto/sboxU

Send me an email (leo.perrin@inria.fr) if you want to join the sboxU mailing list.

Thank you!