Cryptanalysis of a Theorem

Decomposing the Only Known Solution to the Big APN Problem

Alex Biryukov[1] Léo Perrin[1] Aleksei Udovenko[1]

[1]University of Luxembourg, SnT

August 17, 2016

UNIVERSITÉ DU LUXEMBOURG

securityandtrust.lu

Outline

Plan

Definition (DDT)

The <u>DDT</u> of $f : \{0,1\}^n \rightarrow \{0,1\}^n$ is a $2^n \times 2^n$ table such that

$$DDT_f[a, b] = \#\big\{x \in \{0,1\}^n,\ f(x) \oplus f(x \oplus a) = b\big\}.$$

Definition (DDT)

The DDT of $f : \{0,1\}^n \rightarrow \{0,1\}^n$ is a $2^n \times 2^n$ table such that

$$DDT_f[a, b] = \#\{x \in \{0,1\}^n,\ f(x) \oplus f(x \oplus a) = b\}.$$

Definition (APN)

$f : \{0,1\}^n \rightarrow \{0,1\}^n$ is called APN if and only if

$$DDT_f[a, b] \ \leq \ 2 \text{ for all } a \neq 0, b.$$

In other words: the DDT only contains 0 and 2.

Definition (DDT)

The DDT of $f : \{0,1\}^n \rightarrow \{0,1\}^n$ is a $2^n \times 2^n$ table such that

$$DDT_f[a, b] = \#\{x \in \{0,1\}^n,\ f(x) \oplus f(x \oplus a) = b\}.$$

Definition (APN)

$f : \{0,1\}^n \rightarrow \{0,1\}^n$ is called APN if and only if

$$DDT_f[a, b] \ \leq\ 2 \text{ for all } a \neq 0, b.$$

In other words: the DDT only contains 0 and 2.

The Big APN Problem

Does there exist an APN permutation on $GF(2^n)$ if n is even?

Definition (DDT)

The DDT of $f : \{0,1\}^n \rightarrow \{0,1\}^n$ is a $2^n \times 2^n$ table such that

$$DDT_f[a,b] = \#\{x \in \{0,1\}^n,\ f(x) \oplus f(x \oplus a) = b\}.$$

Definition (APN)

$f : \{0,1\}^n \rightarrow \{0,1\}^n$ is called APN if and only if

$$DDT_f[a,b] \ \leq \ 2 \text{ for all } a \neq 0, b.$$

In other words: the DDT only contains 0 and 2.

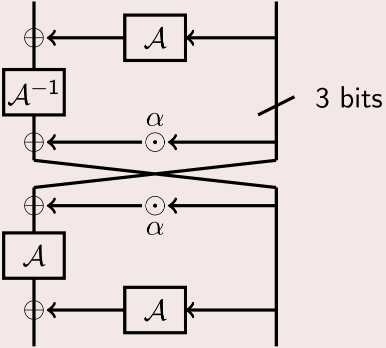
The Big APN Problem

Does there exist an APN permutation on $GF(2^n)$ if n is even?

For $n = 6$, yes! [Dillon et al., 2009]

Our Decomposition (and Main Theorem)

The APN permutation of Dillon et al. is affine-equivalent to...



- for any 3-bit APN permutation $\mathcal{A}$ (e.g. $x \mapsto x^3$)
- for any α such that $\mathrm{Tr}(\alpha) = 0, \alpha \neq 0$.

Plan

S-Box Reverse-Engineering

Definition

Using only the look-up table, *reverse-engineering an S-Box* means recovering unpublished information, e.g.:

- what properties were optimized?
- what structure was used to build it?

S-Box Reverse-Engineering

Definition

Using only the look-up table, *reverse-engineering an S-Box* means recovering unpublished information, e.g.:

- what properties were optimized?
- what structure was used to build it?

Possible Targets

- S-Box of Skipjack [BP, CRYPTO2015]
- S-Box of Streebog/Kuznechik, [BPU, EUROCRYPT2016]
- ...
- The Dillon permutation!

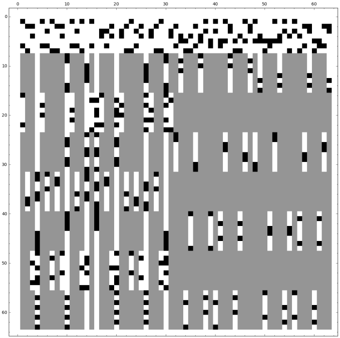
Linear Approximation Table (LAT)

Definition (LAT, Fourier Transform, Walsh Spectrum)

The LAT of $f : \{0,1\}^n \rightarrow \{0,1\}^n$ is a $2^n \times 2^n$ matrix $\mathcal{L}$ where

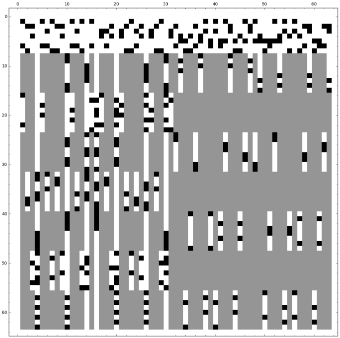
$$\mathcal{L}[a,b] \ = \ \#\{x \in \mathbb{F}_2^n, a \cdot x = b \cdot f(x)\} - 2^{n-1}.$$

Jackson Pollock



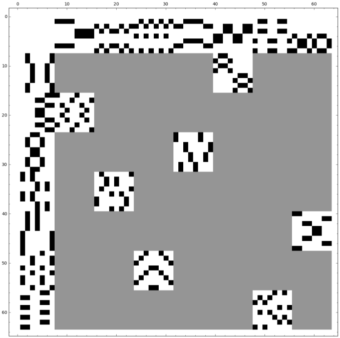
The absolute LAT of S_0.
white=0, grey=4, black=8

Jackson Pollock



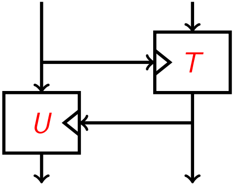
The absolute LAT of S_0.
white=0, grey=4, black=8

$\rightarrow$



The absolute LAT of $\eta \circ S_0$.
η is a linear permutation.

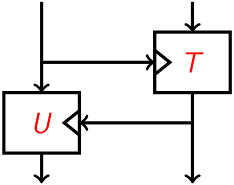
TU-Decomposition



Decomposition of $\eta \circ S_0$.

- T and U are keyed permutations (mini-block ciphers).

TU-Decomposition

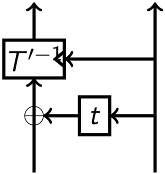


Decomposition of $\eta \circ S_0$.

- T and U are keyed permutations (mini-block ciphers).
- T and U^{-1} are related $\implies$ only attack T.

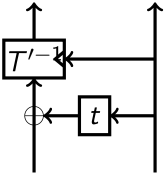
	0	1	2	3	4	5	6	7
T_0	0	6	4	7	3	1	5	2
T_1	7	5	1	6	4	2	0	3
T_2	4	3	2	0	5	6	1	7
T_3	3	5	2	1	4	6	7	0
T_4	1	2	0	6	4	3	7	5
T_5	6	5	2	4	7	0	1	3
T_6	5	2	6	4	0	3	1	7
T_7	2	0	1	6	5	3	4	7

Decomposing T

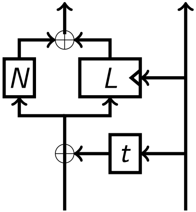


(a) Detaching a linear Feistel round.

Decomposing T

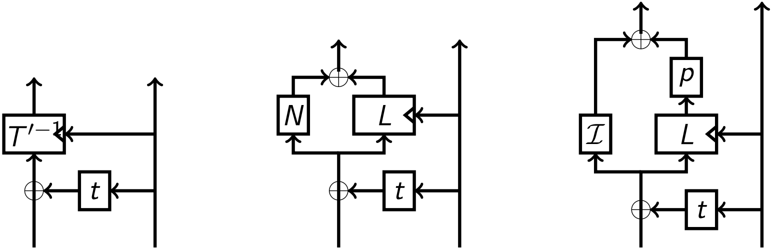


(d) Detaching a linear Feistel round.



(e) Splitting T'^{-1} into N and L.

Decomposing T



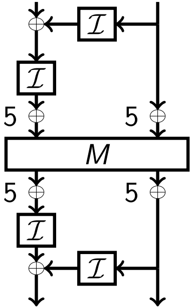
(g) Detaching a linear Feistel round.

(h) Splitting T'^{-1} into N and L.

(i) Simplifying N into $\mathcal{I}$ and linear functions.

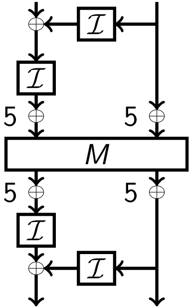
Decomposing T and U

1. Deduce a decomposition (see picture).

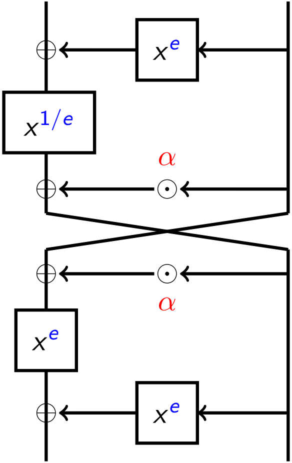


Decomposing T and U

1. Deduce a decomposition (see picture).
2. Get rid of constant additions.
3. Find a nicer representation of M.



Final Decomposition



- Branch size: 3
- $\mathrm{Tr}(\alpha) = 0$
- $e \in \{3, 5, 6\}$

Bit-Sliced Implementation

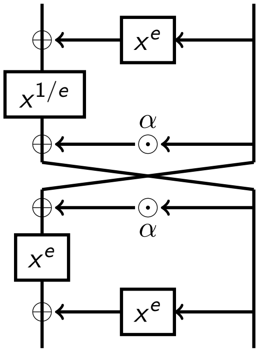
Function $A_0(X_0, \ldots, X_5)$

1. $t = (X_5 \wedge X_3)$
2. $X_0 \oplus= t \oplus (X_5 \wedge X_4)$
3. $X_1 \oplus= t$
4. $X_2 \oplus= (X_4 \vee X_3)$
5. $t = (X_1 \vee X_0)$
6. $X_0 \oplus= (X_2 \wedge X_1) \oplus X_4$
7. $X_1 \oplus= (X_2 \wedge X_0) \oplus X_5 \oplus X_3$
8. $X_2 \oplus= t \oplus X_3$
9. $X_3 \oplus= X_1$
10. $X_4 \oplus= X_2 \oplus X_0$
11. $X_5 \oplus= X_0$
12. $u = X_3$
13. $t = X_4$
14. $X_3 \oplus= t$
15. $X_3 = X_3 \wedge X_5 \oplus t$
16. $X_4 \oplus= ((\neg X_5) \wedge u)$
17. $X_5 \oplus= (t \vee u)$
18. $t = (X_2 \wedge X_0)$
19. $X_3 \oplus= t \oplus (X_2 \wedge X_1)$
20. $X_4 \oplus= t$
21. $X_5 \oplus= (X_1 \vee X_0)$

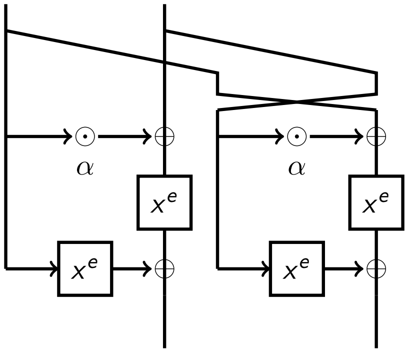
Plan

Definition

- We generalize the structure to any odd branch size:



Open (bijective) butterfly H_e^{α}.



Closed (non-bijective) butterfly V_e^{α}.

CCZ-equivalence

Definition

Two functions are CCZ-equivalent if their graphs are affine-equivalent.

CCZ-equivalence

Definition

Two functions are CCZ-equivalent if their graphs are affine-equivalent.

Theorem

CCZ-equivalence preserves

- *differential uniformity (maximum DDT coefficient),*
- *non-linearity ($\implies$ max coefficient in the LAT).*

CCZ-equivalence

Definition

Two functions are CCZ-equivalent if their graphs are affine-equivalent.

Theorem

CCZ-equivalence preserves

- *differential uniformity (maximum DDT coefficient),*
- *non-linearity ($\implies$ max coefficient in the LAT).*

Lemma

Open and closed butterflies are CCZ-equivalent!

Properties

Theorem (For $\alpha \neq 0, 1$)

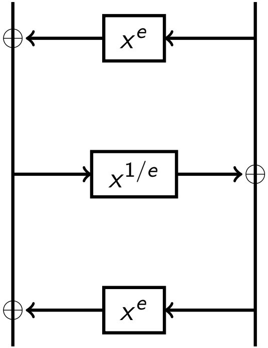
Consider butterflies operating on $2n$ bits with n odd and $e = 3 \times 2^t$.

Differential *The diff. uniformity of V_e^α and H_e^α is at most **4**.*

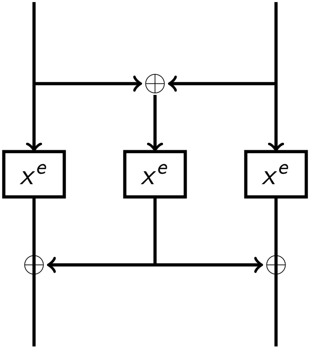
Algebraic $\deg(\mathrm{V}_e^\alpha) = 2$, $\deg(\mathrm{H}_e^\alpha) = n + 1$.

Nonlinearity *(Experimental for small n): $NL(\mathrm{V}_e^\alpha) = NL(\mathrm{H}_e^\alpha) = 2^{2n-1} - 2^n$. The best known to be possible.*

Feistel Network ($\alpha = 1$)



F^e (note $F^e = H_e^1$).



Closed butterfly V_e^1.

Properties of Feistel Butterflies

Theorem (For $\alpha = 1$, i.e. the Feistel case)

Consider butterflies operating on $2n$ bits with n odd and $e = 3 \times 2^t$.

- **Differential** *The diff. uniformity of* V_e^1 *and* H_e^1 *is exactly **4**. The DDT of* V_e^1 *contains only 0 and 4.*
- **Algebraic** $\deg(\mathsf{V}_e^1) = 2$, $\deg(\mathsf{H}_e^1) = n$.

Properties of Feistel Butterflies

Theorem (For $\alpha = 1$, i.e. the Feistel case)

Consider butterflies operating on $2n$ bits with n odd and $e = 3 \times 2^t$.

- Differential: *The diff. uniformity of V_e^1 and H_e^1 is exactly **4**. The DDT of V_e^1 contains only 0 and 4.*
- Algebraic: $\deg(\mathsf{V}_e^1) = 2$, $\deg(\mathsf{H}_e^1) = n$.

Theorem (CCZ-equivalence with a monomial)

Consider butterflies operating on $2n$ bits with n odd and $e = 2^{2k} + 1$

Properties of Feistel Butterflies

Theorem (For $\alpha = 1$, i.e. the Feistel case)

Consider butterflies operating on $2n$ bits with n odd and $e = 3 \times 2^t$.

- **Differential** *The diff. uniformity of V_e^1 and H_e^1 is exactly **4**. The DDT of V_e^1 contains only 0 and 4.*
- **Algebraic** $\deg(V_e^1) = 2$, $\deg(H_e^1) = n$.

Theorem (CCZ-equivalence with a monomial)

Consider butterflies operating on $2n$ bits with n odd and $e = 2^{2k} + 1$

1. V_e^1 *(Lai-Massey-like structure) is Affine-Equivalent to $x \mapsto x^e$ in $\mathbb{F}_2^{2n}$,*
2. H_e^1 *(Feistel Network) is CCZ-equivalent to the same function.*

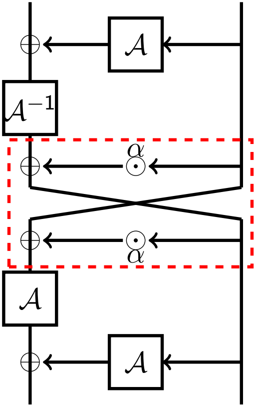
Plan

Flexibility

Consider APN butterflies over 6 bits.

Flexibility

Consider APN butterflies over 6 bits.



- $\mathcal{A}$ can be any APN permutation,
- α can be any element $\neq 0, 1$ with $\mathrm{Tr}(\alpha) = 0$,
- We can XOR any values around the center,
- We can apply identical 3×3 linear permutations on the branches around the center.
- We can swap branches before/after the center (breaks AE but not CCZ-equivalence)

Multiplicative Stability

- For $(a, b) \in (\mathbb{F}_2^n)^2$, $(c, d) \in (\mathbb{F}_2^n)^2$, we define

$$(a, b) \otimes (c, d) \;=\; (ac, bd).$$

Multiplicative Stability

- For $(a, b) \in (\mathbb{F}_2^n)^2$, $(c, d) \in (\mathbb{F}_2^n)^2$, we define

$$(a, b) \otimes (c, d) \; = \; (ac, bd).$$

- For closed butterflies,

$$\mathsf{V}_e^{\alpha}(\lambda x, \lambda y) \; = \; (\lambda^e, \lambda^e) \otimes \mathsf{V}_e^{\alpha}(x, y),$$

- and for open ones:

$$\mathsf{H}_e^{\alpha}(\lambda^e x, \lambda y) \; = \; (\lambda^e, \lambda) \otimes \mathsf{H}_e^{\alpha}(x, y).$$

Parallel Bent Functions

- V_α^3 is affine-equivalent to $(x, y) \mapsto Q(x, y) || Q(y, x)$, with

$$Q(x, y) = x^3(1 + \alpha^2) + x^2 y.$$

Parallel Bent Functions

- V_α^3 is affine-equivalent to $(x, y) \mapsto Q(x, y)||Q(y, x)$, with

$$Q(x, y) = x^3(1 + \alpha^2) + x^2y.$$

- Q is bent (Maiorana-McFarland structure)

Univariate Representation (1/2)

From Dillon et al. (g is their APN permutation):

$$g = f_2 \circ f_1^{-1},$$

where

$$\begin{aligned} f_1(x) &= w^{38}x^{48} + w^{33}x^{40} + w^{28}x^{34} + w^{25}x^{33} + w^{43}x^{32} \\ &+ w^{5}x^{24} + w^{42}x^{20} + x^{17} + w^{2}x^{16} + w^{4}x^{12} \\ &+ w^{7}x^{10} + w^{58}x^{8} + w^{59}x^{6} + w^{5}x^{5} + w^{36}x^{4} \\ &+ w^{47}x^{3} + w^{30}x^{2} + w^{9}x \end{aligned}$$

and

$$\begin{aligned} f_2(x) &= w^{26}x^{48} + w^{60}x^{40} + w^{46}x^{34} + w^{6}x^{33} + w^{61}x^{32} \\ &+ w^{51}x^{24} + w^{53}x^{20} + w^{61}x^{17} + w^{54}x^{16} + w^{55}x^{12} \\ &+ w^{33}x^{10} + w^{33}x^{8} + w^{19}x^{6} + w^{46}x^{5} + w^{51}x^{4} \\ &+ w^{16}x^{3} + w^{37}x^{2} + w^{27}x. \end{aligned}$$

Univariate Representation (2/2)

Other definitions

It still works if we redefine f_1, f_2:

$$\begin{cases} f_1(x) = w^{11}x^{34} + w^{53}x^{20} + x^8 + x, \\ f_2(x) = w^{28}x^{48} + w^{61}x^{34} + w^{12}x^{20} + w^{16}x^8 + x^6 + w^2x. \end{cases}$$

Univariate Representation (2/2)

Other definitions

It still works if we redefine f_1, f_2:

$$\begin{cases} f_1(x) = w^{11}x^{34} + w^{53}x^{20} + x^8 + x, \\ f_2(x) = w^{28}x^{48} + w^{61}x^{34} + w^{12}x^{20} + w^{16}x^8 + x^6 + w^2x. \end{cases}$$

Another decomposition

g is APN if $g = i \circ m \circ i^{-1}$ and either

$$i(x) = w^{37}x^{48} + x^{34} + w^{49}x^{20} + w^{21}x^8 + w^{30}x^6 + x, \; m(x) = x^8,$$

or

$$i(x) = w^{21}x^{34} + x^{20} + x^8 + x, \; m(x) = w^{52}x^8 + w^{36}x.$$

Kim Mapping

Properties

- The "Kim mapping" is the APN function $\kappa(x) = x^3 + x^{10} + wx^{24}$.
- Not a permutation.
- Already known (not found by Dillon et al.).

Kim Mapping

Properties

- The "Kim mapping" is the APN function $\kappa(x) = x^3 + x^{10} + wx^{24}$.
- Not a permutation.
- Already known (not found by Dillon et al.).

$$
\begin{tikzcd}
\text{Dillon permutation} \arrow[d, leftrightarrow, "\text{CCZ-equiv.}"'] \\
\text{Kim mapping}
\end{tikzcd}
$$

Kim Mapping

Properties

- The "Kim mapping" is the APN function $\kappa(x) = x^3 + x^{10} + wx^{24}$.
- Not a permutation.
- Already known (not found by Dillon et al.).

$$
\begin{tikzcd}
\text{Dillon permutation} \arrow[r, leftrightarrow, "\text{affine-equiv.}"] \arrow[d, leftrightarrow, "\text{CCZ-equiv.}"'] & \text{Open Butterfly } \mathrm{H}_3^\alpha \arrow[d, leftrightarrow, "\text{CCZ-equiv.}"'] \\
\text{Kim mapping} \arrow[r, leftrightarrow, "\text{affine-equiv.}"] & \text{Closed Butterfly } \mathrm{V}_3^\alpha
\end{tikzcd}
$$

Plan

Conclusion

There is a Decomposition of the 6-bit APN permutation!

Conclusion

There is a Decomposition of the 6-bit APN permutation!

Open Problems

1. Is the non-linearity of a $2n$-bit butterfly always $2^{2n-1} - 2^n$?

Conclusion

There is a Decomposition of the 6-bit APN permutation!

Open Problems

1. Is the non-linearity of a $2n$-bit butterfly always $2^{2n-1} - 2^n$?
2. **Are there APN Butterflies for $n > 3$?**

Thank you!