Correlation of Quadratic Boolean Functions: Cryptanalysis of All Versions of Full MORUS

Siwei Sun

Joint work with:
Danping Shi Yu Sasaki Chaoyun Li Lei Hu

Chinese Academy of Sciences, China

NTT Secure Platform Laboratories, Japan

imec-COSIC, Dept. Electrical Engineering (ESAT), KU Leuven, Belgium

December 14, 2019

Outlines

1. Correlation and Linear Cryptanalysis
2. Correlation of Quadratic Boolean Functions
3. Cryptanalysis of MORUS
4. Conclusion and Discussion

Outline

Correlation

Let $f : \mathbb{F}_2^n \rightarrow \mathbb{F}_2$ be a Boolean function with ANF

$$f(\boldsymbol{x}) = \sum_{\boldsymbol{u} \in \mathbb{F}_2^n} a_{\boldsymbol{u}} \boldsymbol{x}^{\boldsymbol{u}},$$

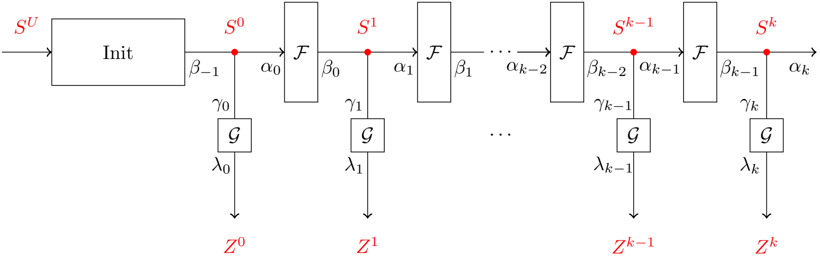
where $\boldsymbol{x} = (x_1, \cdots, x_n)$, $\boldsymbol{u} = (u_1, \cdots, u_n)$, $a_{\boldsymbol{u}} \in \mathbb{F}_2$, and $\boldsymbol{x}^{\boldsymbol{u}} = \prod_{i=1}^{n} x_i^{u_i}$.

Definition (Correlation)

The correlation of an n-variable Boolean function f is $\mathrm{cor}(f) = \frac{1}{2^n} \sum_{\boldsymbol{x} \in \mathbb{F}_2^n} (-1)^{f(\boldsymbol{x})}$, and the weight of the correlation is defined as $-\log_2 |\mathrm{cor}(f)|$.

- $Pr(f = 0) = \frac{1}{2} + \frac{1}{2}\mathrm{cor}(f)$

Linear Cryptanalysis



Object: $\max \left|\text{cor}\left(\sum_{i=0}^{k} \lambda_i Z^i\right)\right|$

Note that $\sum_{i=0}^{k} \lambda_i Z^i$ is a Boolean function whose variables are bits of S^0.

Definition (Correlation)

The correlation of an n-variable Boolean function f is $\mathrm{cor}(f) = \frac{1}{2^n} \sum_{\boldsymbol{x} \in \mathbb{F}_2^n} (-1)^{f(\boldsymbol{x})}$, and the weight of the correlation is defined as $-\log_2 |\mathrm{cor}(f)|$.

- Brute force the input
- Graph-based method [TIM$^+$18]
- ... ...

Outline

Definition (Disjoint Quadratic Boolean Function)

A quadratic Boolean function $f(x_1, \cdots, x_n)$ is disjoint if no variable x_i appears in more than one quadratic term.

Example

$$x_1x_2 + x_3x_4$$

$$x_1x_3 + x_2x_4 + x_2 + x_5$$

Counter-Example

$$x_1x_2 + x_2x_3$$

lemma

Let $f = x_{i_1}x_{i_2} + \cdots + x_{i_{2k-1}}x_{i_{2k}} + x_{j_1} + \cdots + x_{j_s}$ be a disjoint quadratic Boolean function. Then the correlation of f is

$$\begin{cases} (-1)^{\sum_{t=1}^{k} \mathrm{Coe}_f(x_{i_{2t-1}})\mathrm{Coe}_f(x_{i_{2t}})} \cdot 2^{-k} & \{j_1, \cdots, j_s\} \subseteq \{i_1, \cdots, i_{2k}\} \\ 0 & \{j_1, \cdots, j_s\} \subsetneq \{i_1, \cdots, i_{2k}\} \end{cases}$$

where $\mathrm{Coe}_f(\boldsymbol{x}^{\boldsymbol{u}})$ denotes the coefficient of the monomial $\boldsymbol{x}^{\boldsymbol{u}}$ in the ANF of f.

Examples

$$|\mathrm{cor}(x_1x_2 + x_3x_4)| = 2^{-2}$$

$$|\mathrm{cor}(x_1x_3 + x_2x_4 + x_2 + x_5)| = 0$$

$$|\mathrm{cor}(x_1x_3 + x_2x_4 + x_2 + x_3)| = 2^{-2}$$

Idea

Given a quadratic Boolean function, transform it into a disjoint quadratic Boolean function such that the transformation is correlation invariant (up to a minus sign).

Example

$f = x_1x_2 + x_1x_5 + x_2x_3 + x_2x_4 + x_1 + x_2$

Example

$f = x_1x_2 + x_1x_5 + x_2x_3 + x_2x_4 + x_1 + x_2$

$f = x_1{\color{red}x_2} + x_1x_5 + {\color{red}x_2}x_3 + {\color{red}x_2}x_4 + x_1 + x_2$

Example

$f = x_1x_2 + x_1x_5 + x_2x_3 + x_2x_4 + x_1 + x_2$

$f = x_1{\color{red}x_2} + x_1x_5 + {\color{red}x_2}x_3 + {\color{red}x_2}x_4 + x_1 + x_2$

$f = {\color{red}x_2}(x_1 + x_3 + x_4) + x_1x_5 + x_1 + x_2$

Example

$f = x_1x_2 + x_1x_5 + x_2x_3 + x_2x_4 + x_1 + x_2$

$f = x_1x_2 + x_1x_5 + x_2x_3 + x_2x_4 + x_1 + x_2$

$f = x_2(x_1 + x_3 + x_4) + x_1x_5 + x_1 + x_2$

$x_1 \leftarrow x_1 + x_3 + x_4$

$x_j \leftarrow x_j, \quad j \neq 1$

Example

$f = x_1x_2 + x_1x_5 + x_2x_3 + x_2x_4 + x_1 + x_2$

$f = x_1x_2 + x_1x_5 + x_2x_3 + x_2x_4 + x_1 + x_2$

$f = x_2(x_1 + x_3 + x_4) + x_1x_5 + x_1 + x_2$

$x_1 \leftarrow x_1 + x_3 + x_4$

$x_j \leftarrow x_j, \quad j \neq 1$

$f = x_1x_2 + x_1x_5 + x_3x_5 + x_4x_5 + x_1 + x_3 + x_4 + x_2$

Example

$f = x_1x_2 + x_1x_5 + x_2x_3 + x_2x_4 + x_1 + x_2$

$f = x_1x_2 + x_1x_5 + x_2x_3 + x_2x_4 + x_1 + x_2$

$f = x_2(x_1 + x_3 + x_4) + x_1x_5 + x_1 + x_2$

$x_1 \leftarrow x_1 + x_3 + x_4$

$x_j \leftarrow x_j, \quad j \neq 1$

$f = x_1x_2 + x_1x_5 + x_3x_5 + x_4x_5 + x_1 + x_3 + x_4 + x_2$

Example

$$f = x_1x_2 + x_1x_5 + x_2x_3 + x_2x_4 + x_1 + x_2$$

$$f = x_1x_2 + x_1x_5 + x_2x_3 + x_2x_4 + x_1 + x_2$$

$$f = x_2(x_1 + x_3 + x_4) + x_1x_5 + x_1 + x_2$$

$$x_1 \leftarrow x_1 + x_3 + x_4$$

$$x_j \leftarrow x_j, \quad j \neq 1$$

$$f = x_1x_2 + x_1x_5 + x_3x_5 + x_4x_5 + x_1 + x_3 + x_4 + x_2$$

$$f = x_1(x_2 + x_5) + x_3x_5 + x_4x_5 + x_1 + x_3 + x_4 + x_2$$

Example

$$f = x_1x_2 + x_1x_5 + x_2x_3 + x_2x_4 + x_1 + x_2$$
$$f = x_1\textcolor{red}{x_2} + x_1x_5 + \textcolor{red}{x_2}x_3 + \textcolor{red}{x_2}x_4 + x_1 + x_2$$
$$f = \textcolor{red}{x_2}(x_1 + x_3 + x_4) + x_1x_5 + x_1 + x_2$$
$$x_1 \leftarrow x_1 + x_3 + x_4$$
$$x_j \leftarrow x_j, \quad j \neq 1$$
$$f = \textcolor{red}{x_1}x_2 + \textcolor{red}{x_1}x_5 + x_3x_5 + x_4x_5 + x_1 + x_3 + x_4 + x_2$$
$$f = \textcolor{red}{x_1}(x_2 + x_5) + x_3x_5 + x_4x_5 + x_1 + x_3 + x_4 + x_2$$
$$x_2 \leftarrow x_2 + x_5$$
$$x_j \leftarrow x_j, \quad j \neq 2$$

Example

$$f = x_1x_2 + x_1x_5 + x_2x_3 + x_2x_4 + x_1 + x_2$$
$$f = x_1{\color{red}x_2} + x_1x_5 + {\color{red}x_2}x_3 + {\color{red}x_2}x_4 + x_1 + x_2$$
$$f = {\color{red}x_2}(x_1 + x_3 + x_4) + x_1x_5 + x_1 + x_2$$
$$x_1 \leftarrow x_1 + x_3 + x_4$$
$$x_j \leftarrow x_j, \quad j \neq 1$$
$$f = {\color{red}x_1}x_2 + {\color{red}x_1}x_5 + x_3x_5 + x_4x_5 + x_1 + x_3 + x_4 + x_2$$
$$f = {\color{red}x_1}(x_2 + x_5) + x_3x_5 + x_4x_5 + x_1 + x_3 + x_4 + x_2$$
$$x_2 \leftarrow x_2 + x_5$$
$$x_j \leftarrow x_j, \quad j \neq 2$$
$$f = x_1x_2 + x_3x_5 + x_4x_5 + x_1 + x_2 + x_3 + x_4 + x_5$$

Example

$$f = x_1x_2 + x_1x_5 + x_2x_3 + x_2x_4 + x_1 + x_2$$
$$f = x_1x_2 + x_1x_5 + x_2x_3 + x_2x_4 + x_1 + x_2$$
$$f = x_2(x_1 + x_3 + x_4) + x_1x_5 + x_1 + x_2$$
$$x_1 \leftarrow x_1 + x_3 + x_4$$
$$x_j \leftarrow x_j, \quad j \neq 1$$
$$f = x_1x_2 + x_1x_5 + x_3x_5 + x_4x_5 + x_1 + x_3 + x_4 + x_2$$
$$f = x_1(x_2 + x_5) + x_3x_5 + x_4x_5 + x_1 + x_3 + x_4 + x_2$$
$$x_2 \leftarrow x_2 + x_5$$
$$x_j \leftarrow x_j, \quad j \neq 2$$
$$f = x_1x_2 + x_3x_5 + x_4x_5 + x_1 + x_2 + x_3 + x_4 + x_5$$

Example

$$f = x_1x_2 + x_1x_5 + x_2x_3 + x_2x_4 + x_1 + x_2$$
$$f = x_1\color{red}{x_2} + x_1x_5 + \color{red}{x_2}x_3 + \color{red}{x_2}x_4 + x_1 + x_2$$
$$f = \color{red}{x_2}(x_1 + x_3 + x_4) + x_1x_5 + x_1 + x_2$$
$$x_1 \leftarrow x_1 + x_3 + x_4$$
$$x_j \leftarrow x_j, \quad j \neq 1$$
$$f = \color{red}{x_1}x_2 + \color{red}{x_1}x_5 + x_3x_5 + x_4x_5 + x_1 + x_3 + x_4 + x_2$$
$$f = \color{red}{x_1}(x_2 + x_5) + x_3x_5 + x_4x_5 + x_1 + x_3 + x_4 + x_2$$
$$x_2 \leftarrow x_2 + x_5$$
$$x_j \leftarrow x_j, \quad j \neq 2$$
$$f = \color{blue}{x_1x_2} + x_3\color{red}{x_5} + x_4\color{red}{x_5} + x_1 + x_2 + x_3 + x_4 + x_5$$
$$f = \color{blue}{x_1x_2} + \color{red}{x_5}(x_3 + x_4) + x_1 + x_2 + x_3 + x_4 + x_5$$

Example

$$f = x_1x_2 + x_1x_5 + x_2x_3 + x_2x_4 + x_1 + x_2$$
$$f = x_1x_2 + x_1x_5 + x_2x_3 + x_2x_4 + x_1 + x_2$$
$$f = x_2(x_1 + x_3 + x_4) + x_1x_5 + x_1 + x_2$$
$$x_1 \leftarrow x_1 + x_3 + x_4$$
$$x_j \leftarrow x_j, \quad j \neq 1$$
$$f = x_1x_2 + x_1x_5 + x_3x_5 + x_4x_5 + x_1 + x_3 + x_4 + x_2$$
$$f = x_1(x_2 + x_5) + x_3x_5 + x_4x_5 + x_1 + x_3 + x_4 + x_2$$
$$x_2 \leftarrow x_2 + x_5$$
$$x_j \leftarrow x_j, \quad j \neq 2$$
$$f = x_1x_2 + x_3x_5 + x_4x_5 + x_1 + x_2 + x_3 + x_4 + x_5$$
$$f = x_1x_2 + x_5(x_3 + x_4) + x_1 + x_2 + x_3 + x_4 + x_5$$
$$x_3 \leftarrow x_3 + x_4$$
$$x_j \leftarrow x_j, \quad j \neq 3$$
$$f \leftarrow x_1x_2 + x_3x_5 + x_1 + x_2 + x_3 + x_5$$

Theorem

Given a quadratic boolean function $f(\boldsymbol{x}) = f(x_1, \cdots, x_n)$, the algorithm outputs a disjoint quadratic Boolean function $\hat{f}(\boldsymbol{x})$ and an invertible $n \times n$ matrix M, such that $\hat{f}(\boldsymbol{x}) = f(\boldsymbol{x}M)$. Moreover, The algorithm has time complexity $\mathcal{O}(n^{3.8})$ and memory complexity $\Omega(n^2)$.

Remark

On 22-06-2019, we received an E-mail from Ryan Williams (MIT), which indicated that essentially the same theory concerning quadratic forms had been developed much earlier (despite some superficial differences in the appearance).

- Leonard Carlitz: *Gauss sums over finite fields of order* 2^n. Acta Arithmetica. 1969.
- Andrzej Ehrenfeucht and Marek Karpinski: *The computational complexity of (xor, and)-counting problems*. International Computer Science Inst. 1990
- Roland Mirwald and Claus-Peter Schnorr: *The Multiplicative Complexity of Quadratic Boolean Forms*. Theor. Comput. Sci. 1992.

Outline

The CAESAR Competition

- R1: 58 candidates, 2014.3-2015.7
- R2: 29 candidates, 2015.7-2016.8
- R3: 15 candidates, 2016.8-2018.3
- RF: 7 candidates, 2018.3-2019.3

Finalists of CAESAR

Lightweight applications

- ACORN
- ASCON

High-performance applications

- AEGIS
- OCB
- MORUS

Defense in depth

- COLM
- Deoxys-II

6 winners were announce on March 20, 2019.

MORUS

- Designers: Hongjun Wu and Tao Huang
- Stream-cipher like design
- MORUS-640, 128-bit key
- MORUS-1280, 128-bit or 256-bit key
- MORUS-1280-256 was broken in ASIACRYPT 2018 [AEL+18]

Name	State size ($5q$)	Register size (q)	Word size ($q/4$)	Key size
MORUS-640-128	640	128	32	128
MORUS-1280-128	1280	256	64	128
MORUS-1280-256	1280	256	64	256

Encryption Algorithm

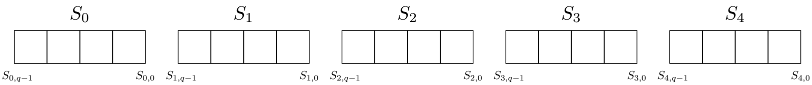


Figure: Internal State

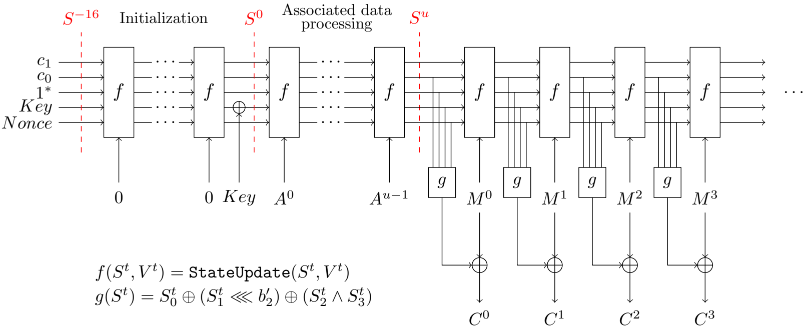
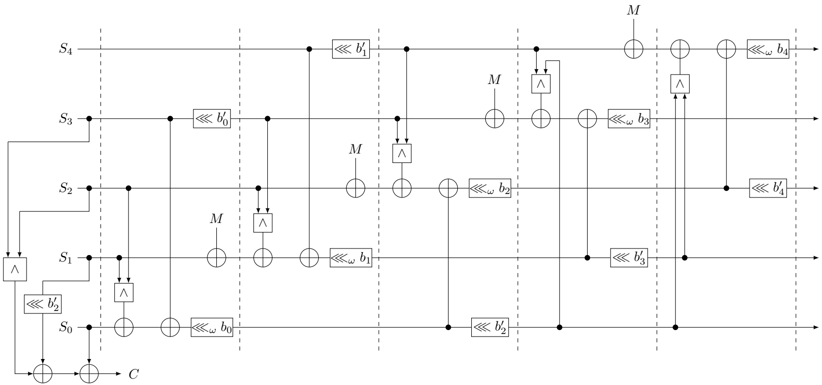
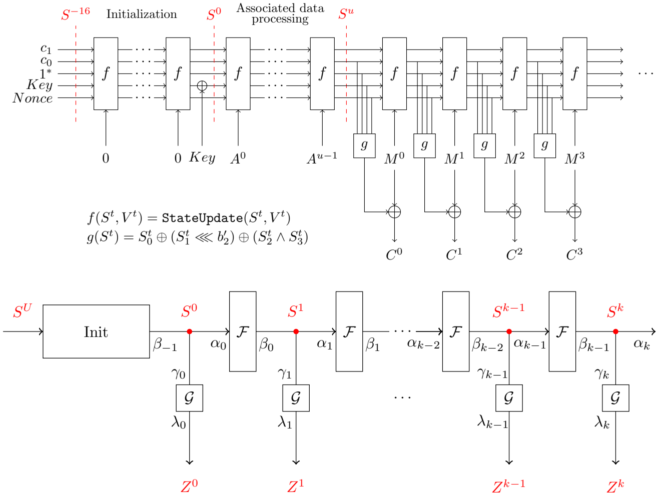


Figure: The encryption algorithm of MORUS

State Update Function

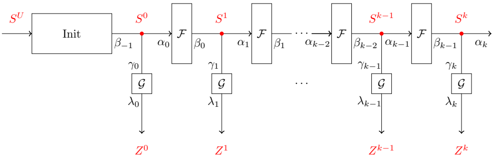


b' is multiple of word size



S^{-16}
Initialization
S^0
Associated data processing
S^u
c_1
c_0
1^*
Key
$Nonce$
f
g
0
0
Key
A^0
A^{u-1}
M^0
M^1
M^2
M^3
C^0
C^1
C^2
C^3
$f(S^t, V^t) = \texttt{StateUpdate}(S^t, V^t)$
$g(S^t) = S_0^t \oplus (S_1^t \lll b_2') \oplus (S_2^t \wedge S_3^t)$
S^U
Init
S^0
S^1
S^{k-1}
S^k
$\mathcal{F}$
$\mathcal{G}$
β_{-1}
α_0
β_0
α_1
β_1
α_{k-2}
β_{k-2}
α_{k-1}
β_{k-1}
α_k
γ_0
γ_1
γ_{k-1}
γ_k
λ_0
λ_1
λ_{k-1}
λ_k
Z^0
Z^1
Z^{k-1}
Z^k

- For a block cipher, we have many tools (Matsui's branch and bound, MILP, SAT, SMT, CP etc.) to search for its linear trails.
- For the key stream generator?



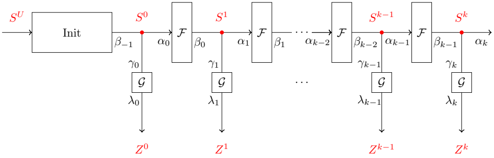
Definition

linear trail A linear trail of the key stream generator shown in Fig:

$$(\beta_{-1}, \gamma_0, \lambda_0, \alpha_0, \beta_0, \cdots, \alpha_{k-1}, \beta_{k-1}, \gamma_k, \lambda_k, \alpha_k)$$

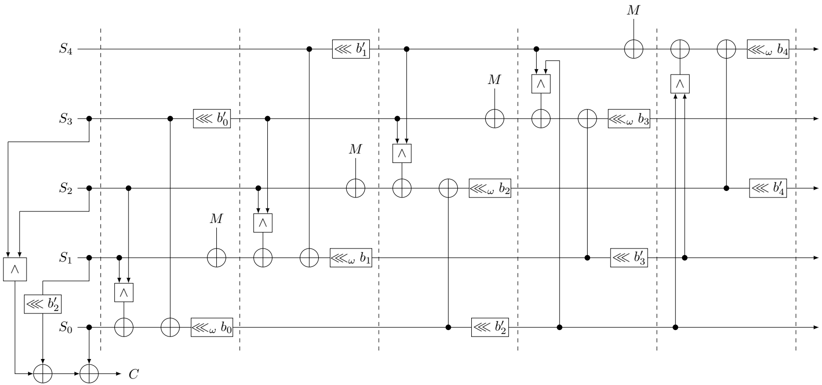
is said to be exploitable if and only if $\beta_{-1} = 0$, $\alpha_k = 0$, and $\alpha_i \oplus \gamma_i \oplus \beta_{i-1} = 0$ for $0 \le i \le k$.

Linear characteristic



$$\begin{cases} \beta_{-1} = 0 \\ \alpha_k = 0 \\ \alpha_i + \gamma_i + \beta_{i-1} = 0, & 0 \leq i \leq k \\ \gamma_i S^i + \lambda_i Z^i = 0, & 0 \leq i \leq k \\ \alpha_i S^i + \beta_i S^{i+1} = 0, & 0 \leq i \leq k-1 \end{cases} \tag{1}$$

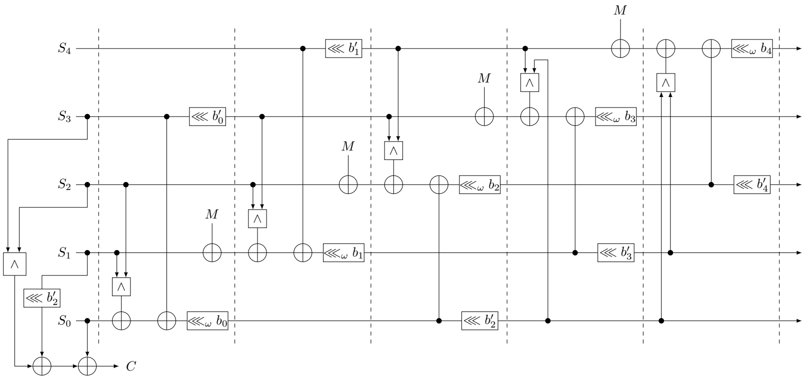
Rotationally Invariant Masks [AEL+18]



b' is multiple of word size

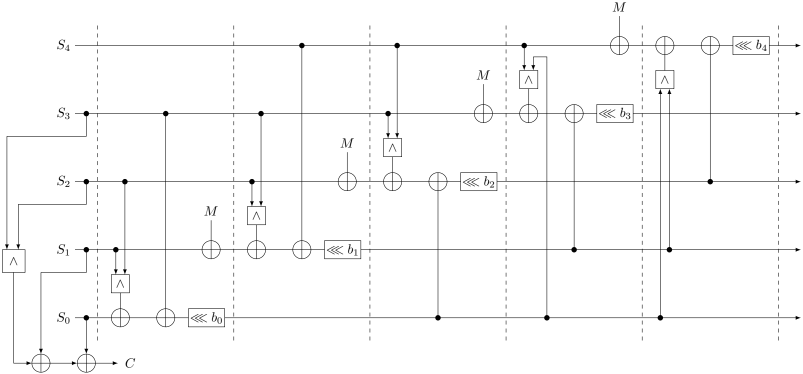
- MiniMORUS: each register contains a single word

MiniMORUS



S_4
S_3
S_2
S_1
S_0
M
M
M
M
$\lll b_0'$
$\lll b_1'$
$\lll b_2'$
$\lll b_3'$
$\lll b_4'$
$\lll_\omega b_0$
$\lll_\omega b_1$
$\lll_\omega b_2$
$\lll_\omega b_3$
$\lll_\omega b_4$
C

MiniMORUS



- Any linear characteristic search tool (Matsui, MILP, SAT/SMT, CP, etc.) can be applied.
- The resulting characteristics are only locally sound!
- Any characteristic can be converted to a quadratic boolean function in variables $S_{i,j}^t$, from which the correlation should be recalculated!

$$\begin{cases} f_1(x_1, x_2, x_3) = x_1x_2 + x_2, \mathrm{cor}(f_1) = 2^{-1} \\ f_2(x_1, x_2, x_3) = x_1x_3, \mathrm{cor}(f_2) = 2^{-1} \end{cases}.$$

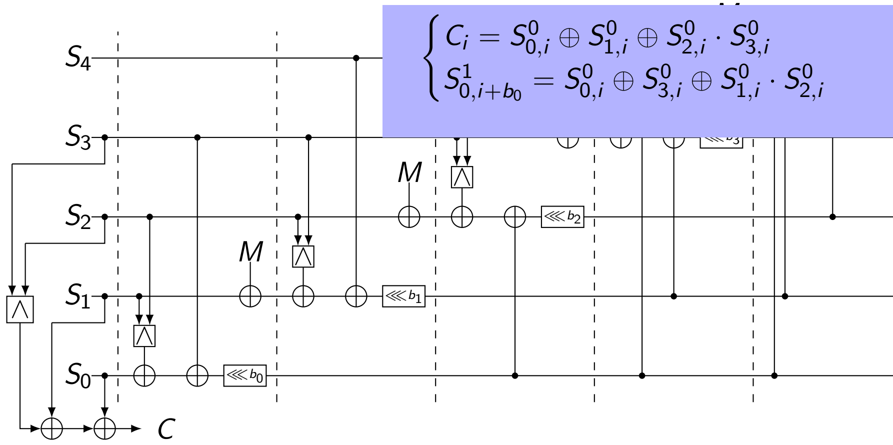
$$f = f_1 + f_2 = x_1x_2 + x_1x_3 + x_2$$

$$\mathrm{cor}(f) = 0 \neq 2^{-2}$$

Table: An invalid trail of MiniMORUS-640 with span 3

Round		Linear masks				
0	α_0	40400000	40400000	00000000	40400000	00000000
		08000008	00400000	00000000	00000000	00000000
		08000008	00200000	00000000	00000000	00400000
		08000008	00200000	00000000	00000000	00400000
		08000008	00200000	00000000	00000000	00400000
	β_0	08000008	00200000	00400000	00000000	00000008
	γ_0	40400000	40400000	00000000	40400000	00000000
	λ_0	40400000				
1	α_1	20600000	28400008	00400000	20600000	00000008
		0c000004	08000008	00000000	00000000	00000008
		0c000004	04000004	08000000	00000000	08000000
		04000004	04000004	00000004	00000000	00000000
		04000004	04000004	00000004	00000000	00000000
	β_1	04000004	04000004	00000004	00000000	00000000
	γ_1	28600008	28600008	00000000	20600000	00000000
	λ_1	28600008				
2	γ_2	04000004	04000004	00000004	00000000	00000000
	λ_2	04000004				

Dependent AND Gates



$$\begin{cases} C_i = S_{0,i}^0 \oplus S_{1,i}^0 \oplus S_{2,i}^0 \cdot S_{3,i}^0 \\ S_{0,i+b_0}^1 = S_{0,i}^0 \oplus S_{3,i}^0 \oplus S_{1,i}^0 \cdot S_{2,i}^0 \end{cases}$$

Table: A linear trail of MiniMORUS-640 with correlation -2^{-8}

Round		Linear masks				
0	α_0	10000000	10000000	00000000	10000000	00000000
		00000002	00000000	00000000	00000000	00000000
		00000002	00000000	00000000	00000000	00000000
		00000002	00000000	00000000	00000000	00000000
		00000002	00000000	00000000	00000000	00000000
	β_0	00000002	00000000	00000000	00000000	00000000
	γ_0	10000000	10000000	00000000	10000000	00000000
	λ_0	10000000				
1	α_1	08000200	08000202	00000002	08000200	00000000
		00004001	00000002	00000002	00000000	00000000
		00004001	00000001	00000000	00000000	00000002
		00004001	00000001	00000000	00000000	00000002
		00004001	00000001	00000000	00000000	00000002
	β_1	00004003	00000003	00000002	00000000	00004000
	γ_1	08000202	08000202	00000002	08000200	00000000
	λ_1	08000202				
2	α_2	00000100	00004100	00000000	00000100	00004000
		00002000	00004000	00000000	00000000	00004000
		00002000	00002000	00000000	00000000	00000000
		00002000	00002000	00000000	00000000	00000000
		00002000	00002000	00000000	00000000	00000000
	β_2	00002000	00002000	00000000	00000000	00000000
	γ_2	00004103	00004103	00000002	00000100	00000000
	λ_2	00004103				
3	γ_3	00002000	00002000	00000000	00000000	00000000
	λ_3	00002000				

- We only list the values for α_i, β_i, γ_i, and λ_i. Actually, for every input and output bits of all the AND gates involved, the solution specifies their masks.
- For every AND gate whose output mask is 1 (active AND gates), we can write down a equation in $S_{i,j}^t$.
- Summing up this equations gives $\sum \lambda_i Z_i$ expressed in a quadratic Boolean function in $S_{i,j}^t$.
- Trails for MiniMORUS can be extended to full MORUS.

Table: A summary of the results

Target	Span	$\lvert\mathrm{cor}\rvert$	Data	Time	Source
MiniMORUS-640	5	2^{-16}	2^{32}	2^{32}	[AEL$^+$18]
	4	2^{-8}	2^{16}	2^{16}	Ours
MiniMORUS-1280	5	2^{-16}	2^{32}	2^{32}	[AEL$^+$18]
	4	2^{-8}	2^{16}	2^{16}	Ours
MORUS-640-128	4	2^{-38}	2^{76}	2^{76}	Ours
MORUS-1280-128	4	2^{-38}	2^{76}	2^{76}	Ours
MORUS-1280-256	5	2^{-76}	2^{152}	2^{152}	[AEL$^+$18]
	4	2^{-38}	2^{76}	2^{76}	Ours

- Distinguishing attack
- Message recovery attack

Assumptions

- S^0 is random (quite reasonable!).
- S^is are independent for different i. (??)

Table: Verification for MiniMORUS

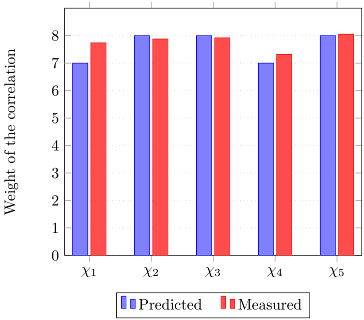
Version	Experiments	Theoretically
MiniMORUS-640	$2^{-7.7919}$	2^{-8}
MiniMORUS-1280	$2^{-8.1528}$	2^{-8}

Table: The five trail fragments of MORUS-640

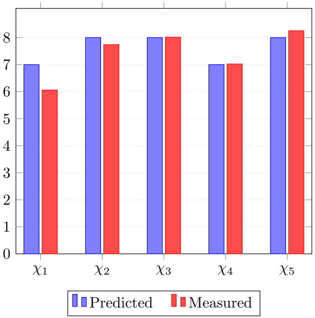
	Trail fragment	Weight
χ_1	$C^0_{\{124,92,60,28\}} \oplus C^1_{\{97,65,33,1\}} = S^1_{4,\{97,65,33,1\}} \oplus S^2_{1,\{96,64,32,0\}}$	7
χ_2	$C^1_{\{123,91,59,27\}} \oplus C^2_{\{96,64,32,0\}} = S^2_{1,\{96,64,32,0\}}$	8
χ_3	$C^2_{\{104,72,40,8\}} \oplus C^3_{\{109,77,45,13\}} = S^3_{1,\{109,77,45,13\}}$	8
χ_4	$C^1_{\{105,73,41,9\}} \oplus C^2_{\{110,78,46,14\}} = S^3_{1,\{109,77,45,13\}} \oplus S^2_{4,\{110,78,46,14\}}$	7
χ_5	$C^2_{\{97,65,33,1\}} = S^1_{4,\{97,65,33,1\}} \oplus S^2_{4,\{110,78,46,14\}}$	8

Table: The five trail fragments of MORUS-1280

	Trail fragment	Weight
χ_1	$C^0_{\{208,144,80,16\}} \oplus C^1_{\{221,157,93,29\}} = S^1_{4,\{221,157,93,29\}} \oplus S^2_{1,\{203,139,75,11\}}$	7
χ_2	$C^1_{\{254,190,126,62\}} \oplus C^2_{\{203,139,75,11\}} = S^2_{1,\{203,139,75,11\}}$	8
χ_3	$C^2_{\{194,130,66,2\}} \oplus C^3_{\{207,143,79,15\}} = S^3_{1,\{207,143,79,15\}}$	8
χ_4	$C^1_{\{212,148,84,20\}} \oplus C^2_{\{225,161,97,33\}} = S^3_{1,\{207,143,79,15\}} \oplus S^2_{4,\{225,161,97,33\}}$	7
χ_5	$C^2_{\{221,157,93,29\}} = S^1_{4,\{221,157,93,29\}} \oplus S^2_{4,\{225,161,97,33\}}$	8



(a) MORUS-640



(b) MORUS-1280

Figure: Experimental verification of the trail fragments of MORUS-640 and MORUS-1280

Outline

- Correlation of quadratic Boolean function can be computed efficiently.
- How about Boolean functions with higher degrees?
- How can we search for trails which are not rotationally invariant?
- MILP based search can only deal with small spans.
- Some manual analysis targeting Trivium, SNOW, and ZUC using very large spans!

Thanks! Any questions?

References I

Tomer Ashur, Maria Eichlseder, Martin M. Lauridsen, Gaëtan Leurent, Brice Minaud, Yann Rotella, Yu Sasaki, and Benoît Viguier.
Cryptanalysis of MORUS.
In *Advances in Cryptology - ASIACRYPT 2018 - 24th International Conference on the Theory and Application of Cryptology and Information Security, Brisbane, QLD, Australia, December 2-6, 2018, Proceedings, Part II*, pages 35–64, 2018.

Yosuke Todo, Takanori Isobe, Willi Meier, Kazumaro Aoki, and Bin Zhang.
Fast Correlation Attack Revisited - Cryptanalysis on Full Grain-128a, Grain-128, and Grain-v1.
In *Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part II*, pages 129–159, 2018.