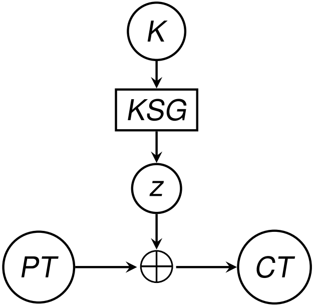
Semi-bent Boolean Functions Arising from CA

Luca Mariot, Martina Saletta, Alberto Leporati, Luca Manzoni

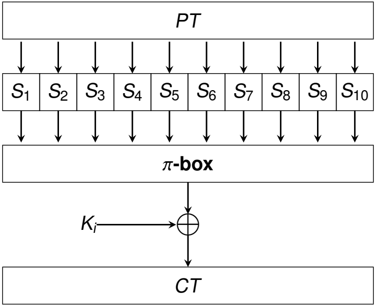
L.Mariot@tudelft.nl

ACRI 2020 – December 2-4, 2020

Boolean Functions in Symmetric Ciphers



(a) Stream cipher



(b) Block cipher

Boolean functions $f : \{0,1\}^n \to \{0,1\}$ are used in:

- *Stream ciphers*, to design the *keystream generator* (KSG)
- *Block ciphers*, as the coordinate functions of *S-boxes* (S_i)

Boolean Functions - Basic Representations

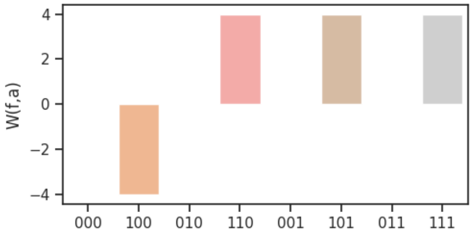
- Truth table: a 2^n-bit vector Ω_f specifying $f(x)$ for all $x \in \{0,1\}^n$

(x_1, x_2, x_3)	000	100	010	110	001	101	011	111
Ω_f	0	1	1	0	1	0	1	0

- Algebraic Normal Form (ANF): Sum (XOR) of products (AND)

$$f(x_1, x_2, x_3) = x_1 \oplus x_2 \oplus x_3 \oplus x_2 x_3$$

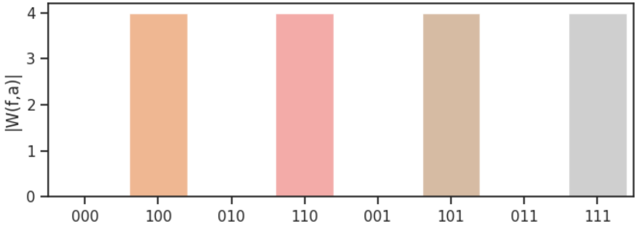
- Walsh Transform: correlation with linear functions $a \cdot x$,
  $W(f,a) = \sum_{x \in \{0,1\}^n} (-1)^{f(x) \oplus a \cdot x}$ for all $a \in \{0,1\}^n$



Special classes of functions

Boolean functions with as "flat" as possible Walsh spectrum:

- Bent functions: $W(f,a) = \pm 2^{\frac{n}{2}}$ for all $a \in \{0,1\}^n$
  - Reach the highest possible nonlinearity
  - Exist only for n even and they are *unbalanced*
- Semi-bent functions: $W(f,a) = \begin{cases} 0, \pm 2^{\frac{n+1}{2}} & \text{if } n \text{ odd}, \\ 0, \pm 2^{\frac{n+2}{2}} & \text{if } n \text{ even} \end{cases}$
  - Can be balanced, and exist for every n
  - good trade-off of nonlinearity and other properties
- Plateaued functions: $W(f,a) \in \{-2^r, 0, 2^r\}$



Constructions of good Boolean Functions

- Number of Boolean functions of n variables: 2^{2^n}
- $\Rightarrow$ too huge for exhaustive search when $n > 5$!

In practice, one can resort to *algebraic constructions*

- *Primary constructions*: (semi-)bent functions are built from scratch (e.g., Maiorana-McFarland construction [M73])
- *Secondary constructions*: new (semi-)bent functions are obtained from existing ones (e.g., Rothaus's construction [R76])

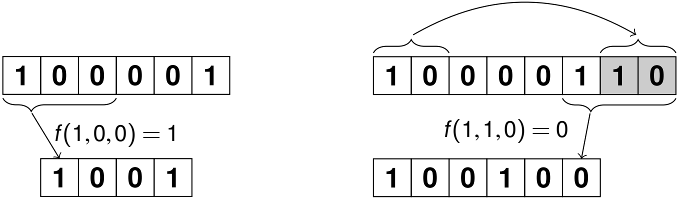
$$\Downarrow$$

Focus of our work: Secondary constructions of semi-bent functions based on Cellular Automata

Cellular Automata (CA)

- Each cell updates its state $s \in \{0,1\}$ by applying a local rule $f : \{0,1\}^m \to \{0,1\}$ to itself and the $m-1$ cells to its right

Example: $n = 6$, $m = 3$, $f(s_i, s_{i+1}, s_{i+2}) = s_i \oplus s_{i+1} \oplus s_{i+2}$

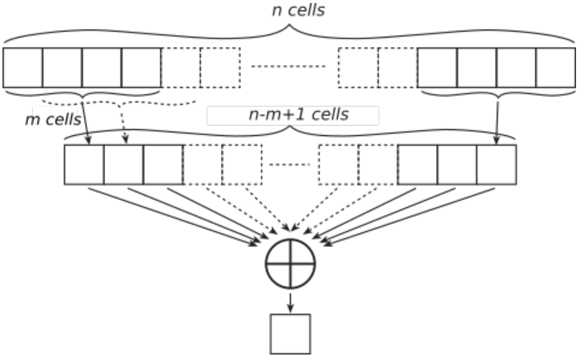


No Boundary CA – NBCA　　Periodic Boundary CA – PBCA

- Uses of PBCA: S-boxes [SS08, MPLJ19], PRNG [W86, LM14]
- Uses of NBCA: Secret Sharing Schemes [MGLF20]

Secondary Construction Scheme

- **Idea:** Define a function $f : \{0,1\}^n \to \{0,1\}$ by XORing all $n - m + 1$ output cells of a NBCA



- Recursive construction parameterized on n, with $n = m$ being the base case (= local rule)

Preservation of Algebraic Degree

Definition

Given a CA of length $n \geq m$ with the local rule $f : \{0,1\}^m \to \{0,1\}$, define $f^* : \{0,1\}^n \to \{0,1\}$ for all $x \in \{0,1\}^n$ as:

$$f^*(x) = \bigoplus_{i=1}^{n-m+1} f(x_i, \cdots, x_{i+m-1}) = f(x_1, \cdots, x_m) \oplus \cdots \oplus f(x_{n-m+1}, \cdots, x_n)$$

Algebraic degree: degree of the ANF of f

$$f(x_1, x_2, x_3) = x_1 \oplus x_2 \oplus x_3 \oplus x_2 x_3 \ \Rightarrow \ \text{degree } 2$$

Lemma

Let $f : \mathbb{F}_2^m \to \mathbb{F}_2$ be a Boolean function of m variables. For any $n \geq m$, the function f^* has the same algebraic degree of f.

Search Strategy

Goal: characterize semi-bent rules f of m variables generating semi-bent functions f^* for all $n \in \mathbb{N}$ under our construction

Empirical search of such rules based on these ideas:

- For each considered rule, check semi-bentness of f^* only up to a certain n (in our case, $n = 20$)
- To reduce the search space, check only *quadratic functions* (degree 2), since they are all plateaued [C10]
- $\Rightarrow$ Combinatorial algorithm to exhaustively search the space of quadratic ANF

Search Algorithm

SEARCH-ANF(m, n, d)

Init: For $1 \leq k \leq d$, build the family $\mathcal{I}_k$ of monomials of degree k, set all 2^m ANF coefficients of f to 0 and initialize $\mathcal{L}$ as the empty list

Loop: For all subsets $\mathcal{T} \subseteq \mathcal{I}_d$ do:

- Init: Reset all d-degree terms in the ANF to 0
- Inst: For all $T \in \mathcal{T}$, set the ANF coefficient a_T to 1
- Loop: For all subsets $\mathcal{P} \subseteq \bigcup_{k=1}^{d-1} \mathcal{I}_k$ do:
  1. Reset all terms of degree less than d to 0
  2. For all $P \in \mathcal{P}$, set a_P to 1
  3. Recover the truth table of f from the ANF
  4. If f is semi-bent, then for $m < i \leq n$ apply the CA construction with i cells
  5. If f^* is semi-bent for all $m < i \leq n$, add f to $\mathcal{L}$

Output: return $\mathcal{L}$

Experiments

- We exhaustively enumerated the space of quadratic local rules of $3 \leq m \leq 6$ variables
- For each m, we further filtered those semi-bent rules that always generate balanced functions

m	2^{2^m}	$S_{m,2}$	QSB	Bal
3	256	56	24	8
4	65 536	1 008	0	0
5	$\approx 4.3 \cdot 10^9$	32 736	2 208	280
6	$\approx 1.84 \cdot 10^{19}$	$2.1 \cdot 10^6$	12 208	1937

Conclusions and Open Problems

Remarkable findings:

- For $m = 4$, our construction always fails. No semi-bent rule generates semi-bent functions of up to 20 variables
- All filtered balanced rules generate semi-bent functions with the same number of linear structures
- For $m = 3$ rule 30 and rule 210 occurred in the filtered set

Open problems:

- Investigate if our construction fails for other values of m
- Theoretical characterization of the rules in the filtered set
- Analyze the periods of spatially periodic preimages in quadratic CA [MLDF17]

References

- [C10] C. Carlet. Boolean Functions for Cryptography and Error Correcting Codes. In Boolean Models and Methods in Mathematics, Computer Science, and Engineering, pages 257–397. Cambridge University Press (2010)
- [LM14] A. Leporati and L. Mariot. Cryptographic properties of bipermutive cellular automata rules. J. Cell. Autom. 9(5-6):437–475 (2014)
- [MGLF20] L. Mariot, M. Gadouleau, E. Formenti, and A. Leporati. Mutually orthogonal latin squares based on cellular automata. Des. Codes Cryptogr. 88(2):391–411 (2020)
- [MLDF17] L. Mariot, A. Leporati, A. Dennunzio, and E. Formenti. Computing the periods of preimages in surjective cellular automata. Nat. Comput. 16(3):367–381 (2017)
- [MPLJ19] L. Mariot, S. Picek, A. Leporati, and D. Jakobovic. Cellular automata based S-boxes. Cryptography and Communications 11(1):41–62 (2019)
- [M73] R. L. McFarland. A family of difference sets in non-cyclic groups. J. Comb. Theory, Ser. A 15(1):1–10 (1973)
- [R76] O. S. Rothaus. On "bent" functions. J. Comb. Theory, Ser. A, 20(3):300–305 (1976)
- [SS08] M. Szaban and F. Seredynski. Cryptographically strong s-boxes based on cellular automata. In ACRI 2008, Proceedings, pages 478–485 (2008)
- [W86] S. Wolfram. Cryptography with cellular automata. In CRYPTO '85, vol. 218 of LNCS, pages 429–432 (1986).