The final publication is available at Springer via https://doi.org/10.1007/978-3-319-44365-2_17

Asynchrony Immune Cellular Automata

Luca Mariot[1,2]

[1] Dipartimento di Informatica, Sistemistica e Comunicazione, Università degli Studi Milano-Bicocca, Viale Sarca 336, 20126 Milano, Italy
[2] Laboratoire I3S, Université Nice-Sophia Antipolis, 2000 Route des Lucioles, 06903 Sophia Antipolis, France
luca.mariot@unimib.it

Abstract. We introduce the notion of asynchrony immunity for cellular automata (CA), which can be considered as a generalization of correlation immunity in the case of boolean functions. The property could have applications in cryptography, namely as a countermeasure for side-channel attacks in CA-based cryptographic primitives. We give some preliminary results about asynchrony immunity, and we perform an exhaustive search of $(3,10)$–asynchrony immune CA rules of neighborhood size 3 and 4. We finally observe that all discovered asynchrony-immune rules are center-permutive, and we conjecture that this holds for any size of the neighborhood.

Keywords: cellular automata · cryptography · asynchrony immunity · correlation immunity · nonlinearity · side-channel attacks · permutivity

1 Introduction

In the last years, research about cryptographic applications of cellular automata (CA) focused on the properties of the underlying local rules [8,7,6]. In fact, designing a CA-based cryptographic primitive using local rules that are not highly nonlinear and correlation immune could make certain attacks more efficient.

The aim of this short paper is to begin investigating a new property related to asynchronous CA called *asynchrony immunity* (AI), which could be of interest in the context of side-channel attacks. This property can be described by a three-move game between a user and an adversary. Let $r,m \in \mathbb{N}$, $n = m+2r$ and $t \leq m$. The game works as follows:

1. The user chooses a local rule $f : \mathbb{F}_2^{2r+1} \to \mathbb{F}_2$ of radius r
2. The adversary chooses $j \leq t$ cells of the CA in the range $\{r, \cdots, m+r\}$.
3. The user evaluates the output distribution D of the CA $F : \mathbb{F}_2^{m+2r} \to \mathbb{F}_2^m$ and the distribution $\tilde{D}$ of the asynchronous CA $\tilde{F} : \mathbb{F}_2^{m+2r} \to \mathbb{F}_2^m$ where the t cells selected by the adversary are not updated
4. *Outcome*: if both D and $\tilde{D}$ equals the uniform distribution, the user wins. Otherwise, the adversary wins

A cellular automaton rule $f : \mathbb{F}_2^{2r+1} \to \mathbb{F}_2$ is called (t,n)–asynchrony immune if, for all lengths $2r < k \leq n$ and for all $j \leq t$, both the asynchronous CA $\tilde{F} : \mathbb{F}_2^k \to \mathbb{F}_2^{k-2r}$

resulting from not updating any subset of j cells and the corresponding synchronous CA $F : \mathbb{F}_2^k \to \mathbb{F}_2^{k-2r}$ are balanced, that is, the cardinality of the counterimage of each k-bit configuration equals 2^{2r}. Thus, asynchrony immune CA rules represent the winning strategies of the user in the game described above.

Notice the difference between the asynchrony immunity game and the *t–resilient functions game* [5]: in the latter, generic vectorial boolean functions $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$ are considered instead of cellular automata, and the adversary selects both values and positions of the t input variables.

The side-channel attack model motivating our work is the following. Suppose that a CA of length n is used as an S-box in a block cipher, and that an attacker is able to inject clock faults by making t cells not updating. If the CA is not (t,n)–AI, then the attacker could gain some information on the internal state of the cipher by analyzing the differences of the output distributions in the original CA and the asynchronous CA.

In the remainder of this paper, we define the considered model of (asynchronous) CA in Section 2, and we formally introduce the definition of asynchrony immunity in Section 3, giving some basic theoretical results regarding this property. In particular, we show that AI is invariant under the operations of reflection and complement. We then perform in Section 4 an exhaustive search of $(3,10)$–asynchrony immune cellular automata up to neighborhood size 4, computing also their nonlinearity values. We finally observe that all discovered rules are center-permutive, and we conjecture that this is a necessary condition for asynchrony immunity.

2 Preliminaries

In this work, we consider one-dimensional CA as a particular kind of *vectorial boolean functions*, i.e. mappings of the type $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$ where $\mathbb{F}_2 = \{0,1\}$ denotes the finite field with two elements. Here we cover only the essential concepts, referring the reader to [4] for further information on vectorial boolean functions.

A vectorial boolean function $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$ (also called an (n,m)–function) is defined by *m coordinate functions* $f_i : \mathbb{F}_2^n \to \mathbb{F}_2$, where for all $x \in \mathbb{F}_2^n$ and $i \in \{0,\cdots,m-1\}$, the value of $f_i(x)$ specifies the output of the $i-th$ bit of F.

Let $r,m \in \mathbb{N}$ be positive integers and $f : \mathbb{F}_2^{2r+1} \to \mathbb{F}_2$ be a boolean function of $2r+1$ variables. The *cellular automaton* of length $n = m+2r$ and local rule f of radius r is the (n,m)–function $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$ defined for all $x = (x_0,\cdots,x_{n-1}) \in \mathbb{F}_2^n$ as:

$$F(x_0,\cdots,x_{n-1}) = (f(x_0,\cdots,x_{2r}), f(x_1,\cdots,x_{2r+1}),\cdots,f(x_m,\cdots,x_{n-1})) \ . \quad (1)$$

Thus, a CA F is defined by the synchronous application of the local rule f to all the central input variables $\{x_r,\cdots,x_{m+r}\}$. This means that, for all $i \in \{0,\cdots,m-1\}$, the $i-th$ coordinate function of F is defined as $f_i(x) = f(x_i,\cdots,x_{i+2r})$.

Let $I = \{i_1,\cdots,i_t\} \subseteq [m] = \{0,\cdots,m-1\}$ be a subset of indices. The *t–asynchronous CA* (t–ACA) $\tilde{F}_I$ induced by I on a CA $F : \mathbb{F}_2^{m+2r} \to \mathbb{F}_2^m$ is obtained by preventing the input variables $x_{i_1+r},\cdots,x_{i_t+r}$ to update. In particular, for all indices $i_k \in I$ the coordinate function f_{i_k} equals the identity, while for the remaining indices $j \in J = [m] \setminus I$ function f_j still corresponds to the local rule f applied to the neighborhood $\{j,\cdots,j+2r\}$.

The final publication is available at Springer via https://doi.org/10.1007/978-3-319-44365-2_17

3 Basic Definition and Properties of Asynchrony Immunity

A CA $F : \mathbb{F}_2^{m+2r} \to \mathbb{F}_2^m$ is *balanced* if for all $y \in \mathbb{F}_2^m$ it holds that $|F^{-1}(y)| = 2^{2r}$. We formally define asynchrony immunity in CA as follows:

Definition 1. *Let $m,n,r,t \in \mathbb{N}$ be positive integers with $n = m+2r$ and $t \leq m$, and let $f : \mathbb{F}_2^{2r+1} \to \mathbb{F}_2$ be a local rule of radius r. Rule f is called* (t,n)–asynchrony immune *((t,n)–AI) if, for all $0 < k \leq m$ and for all sets $I \subseteq [k]$ with $|I| \leq \min\{k,t\}$, both the CA $F : \mathbb{F}_2^{k+2r} \to \mathbb{F}_2^k$ and the t–ACA $\tilde{F}_I : \mathbb{F}_2^{k+2r} \to \mathbb{F}_2^m$ are balanced, i.e. $|F^{-1}(y)| = |\tilde{F}_I^{-1}(y)| = 2^{2r}$ holds for all $y \in \mathbb{F}_2^m$.*

Remark 1. The definition of (t,n)–asynchrony immunity implies in particular that the local rule f is itself balanced, i.e. $|f^{-1}(0)| = |f^{-1}(1)| = 2^{2r}$.

Among all possible $2^{2^{2r+1}}$ rules of radius r, we are interested in finding asynchrony immune rules that satisfy additional useful cryptographic properties, such as high non-linearity. As a consequence, proving necessary conditions for a rule being (t,n)–AI helps one to prune the search space for possible candidates.

We begin by showing that asynchrony immunity is invariant under reflection and complement. To this end, recall that the *reverse* of a vector $x = (x_0, \cdots, x_{n-1})$ is the same vector in reverse order, i.e. $x^R = (x_{n-1}, \cdots, x_0)$, while the *complement* of x is the vector $x^C = (1 \oplus x_0, \cdots, 1 \oplus x_n)$. Given $f : \mathbb{F}_2^{2r+1} \to \mathbb{F}_2$, the *reflected* and *complemented* rules f^R and f^C are respectively defined as $f^R(x) = f(x^R)$ and $f^C(x) = 1 \oplus f(x)$, for all $x \in \mathbb{F}_2^{2r+1}$. For all $m \in \mathbb{N}$, the reflected and complemented CA $F^R : \mathbb{F}_2^{m+2r} \to \mathbb{F}_2^m$ and $F^C : \mathbb{F}_2^{m+2r} \to \mathbb{F}_2^m$ are defined for all $x \in \mathbb{F}_2^n$ as follows:

$$F^R(x) = F(x^R)^R = (f(x_{2r}, \cdots, x_0), \cdots, f(x_{n-1}, \cdots, x_m)) \ , \tag{2}$$

$$F^C(x) = \underline{1} \oplus F(x) = (1 \oplus f(x_0, \cdots, x_{2r}), \cdots, 1 \oplus f(x_m, \cdots, x_{n-1})) \ . \tag{3}$$

The following result shows that asynchrony immunity is preserved under reflection and complement.

Lemma 1. *Let $f : \mathbb{F}_2^{2r+1} \to \mathbb{F}_2$ be a (t,n)–AI local rule, with $n = m+2r$ and $t \leq m$. Then, the reflected and complemented rules f^R and f^C are (t,n)–AI as well.*

Proof. Let $0 < k \leq m$ and $I = \{i_1, \cdots, i_l\} \subseteq [k]$, with $l \leq \min\{k,t\}$.

For the reflected rule f^R, we know by Equation (2) that $F^R(x) = F(x^R)^R$. It follows that the reflection of the l-ACA $\tilde{F}_I$ is defined as:

$$\tilde{F}_I^R(x) = \tilde{F}_J(x^R)^R = (f(x_{2r}, \cdots, x_0), \cdots, x_{j_1}, \cdots, x_{j_l}, \cdots, f(x_{k+2r-1}, \cdots, x_k)) \ , \tag{4}$$

where $J = \{j_1, \cdots, j_l\}$ and $j_s = k - i_s$ for all $1 \leq s \leq l$. Rule f is (t,n)–AI and J is still a set of $l \leq t$ indices, thus $|F^{-1}(y)| = |\tilde{F}_J^{-1}(y)| = 2^{2r}$ for all $y \in \mathbb{F}_2^m$. Since the reverse operator is a bijection over both $\mathbb{F}_2^{k+2r}$ and $\mathbb{F}_2^k$, by Equation (4) *it results that $|(F^R)^{-1}(y)| = |F^{-1}(y)|$ and $|(\tilde{F}_I^R)^{-1}(y)| = |\tilde{F}_J^{-1}(y)|$. Thus, the reflected rule f^R is (t,n)–AI as well.*

Analogously, for the complemented rule f^C the l-ACA $\tilde{F}_I$ is defined as:

$$\tilde{F}_I^C(x) = (1 \oplus f(x_0, \cdots, x_{2r}), \cdots, x_{i_1}, \cdots, x_{i_l}, \cdots, 1 \oplus f(x_k, \cdots, x_{k+2r-1})) \ . \tag{5}$$

Hence we can compute $\tilde{F}_I^C$ by XORing $\tilde{F}_I$ with a bitmask composed of all 1s excepts in the positions $i_1,\cdots,i_l$. Since this operation is again a bijection over $\mathbb{F}_2^k$ and rule f is (t,n)–AI, it means that $|(F^C)^{-1}(y)| = |F^{-1}(y)| = 2^{2r}$ and $|(\tilde{F}_I^C)^{-1}(y)| = |\tilde{F}_I^{-1}(y)| = 2^{2r}$ for all $y \in \mathbb{F}_2^m$. Thus, f^C is also (t,n)–AI. □

4 Search of AI Rules up to 4 Variables

In order to search for asynchrony immune rules having additional cryptographic properties, by Remark 1 and Lemma 1 we only need to explore balanced rules under the equivalence classes induced by reflection and complement. We performed an exhaustive search among all elementary CA rules of radius $r = 1$ in order to find those satisfying (t,n)–asynchrony immunity with $t = 3$ and $n = 10$. The reason why we limited our analysis to these particular values is twofold. First, checking for asynchrony immunity is a computationally cumbersome task, since it requires to determine the output distribution of the t-ACA for all possible choices of at most t blocked cells. Second, the sizes of vectorial boolean functions employed as nonlinear components in several real-world cryptographic primitives, such as KECCAK [2], is not large.

In our quest for AI rules we also took into account the *nonlinearity* property, which is crucial in the design of several cryptographic primitives. Formally, a boolean function is *linear* if it is a linear combination of the input variables. The *nonlinearity* of a boolean function $f : \mathbb{F}_2^n \to \mathbb{F}_2$ is the minimum Hamming distance of f from all linear functions, and it equals $Nl(f) = 2^{-1}(2^n - W_{max}(f))$, where $W_{max}(f)$ is the maximum absolute value of the *Walsh transform* of f [3].

Up to reflection and complement, and neglecting the identity rule that is trivially AI for every length n and order t, we found that only rule 60 is $(3,10)$–asynchrony immune. However, since rule 60 is linear it is not interesting from the cryptographic standpoint. We thus extended the search by considering all local rules of 4 variables defined on an asymmetric neighborhood. The corresponding CA F is defined as:

$$F(x_0,\cdots,x_{n-1}) = (f(x_0,x_1,x_2,x_3),\cdots,f(x_m,x_{m+1},x_{m+2},x_{m+3})) \ . \tag{6}$$

The search returned a total of 18 rules satisfying $(3,10)$–asynchrony immunity, among which several of them were nonlinear. Table 1 reports the Wolfram codes of the discovered rules, along with their nonlinearity values and algebraic normal form (ANF). One can notice from the ANF column in Table 1 that all discovered rules depend on the input variable x_1 in a linear way. This means that each rule can be written as $f(x_0,x_1,x_2,x_3) = x_1 \oplus g(x_0,x_2,x_3)$, where $g : \mathbb{F}_2^3 \to \mathbb{F}_2$. This means that the discovered rules are all *center-permutive*, i.e. by fixing the values of all variables except x_1 the resulting restrictions of the functions are permutations over $\mathbb{F}_2$. Remark that the elementary rule 60 is center permutive as well, being defined as $f(x_0,x_1,x_2) = x_1 \oplus x_2$. This seems to suggest that center-permutivity is a necessary condition for asynchrony immunity, a property that would greatly reduce the search space for possible AI candidates with interesting cryptographic properties. For future research, we thus plan to investigate the following conjecture:

Conjecture 1. Let $f : \mathbb{F}_2^d \to \mathbb{F}_2$ be a (t,n)–asynchrony immune rule of d variables. Then, rule f is center-permutive.

Rule	$Nl(f)$	$f(x_0,x_1,x_2,x_3)$	Rule	$Nl(f)$	$f(x_0,x_1,x_2,x_3)$
13107	0	$1 \oplus x_1$	14028	2	$x_1 \oplus x_0x_3 \oplus x_2x_3 \oplus x_0x_2x_3$
13116	4	$x_1 \oplus x_2 \oplus x_3 \oplus x_2x_3$	14643	2	$1 \oplus x_1 \oplus x_0x_3 \oplus x_0x_2x_3$
13155	2	$1 \oplus x_1 \oplus x_2 \oplus x_0x_2 \oplus x_2x_3 \oplus x_0x_2x_3$	14796	2	$x_1 \oplus x_3 \oplus x_0x_3 \oplus x_0x_2x_3$
13164	2	$x_1 \oplus x_0x_2 \oplus x_3 \oplus x_0x_2x_3$	15411	4	$1 \oplus x_1 \oplus x_3 \oplus x_2x_3$
13203	2	$1 \oplus x_1 \oplus x_0x_2 \oplus x_0x_2x_3$	15420	0	$x_1 \oplus x_2$
13212	2	$x_1 \oplus x_2 \oplus x_0x_2 \oplus x_3 \oplus x_2x_3 \oplus x_0x_2x_3$	15555	0	$1 \oplus x_1 \oplus x_2 \oplus x_3$
13251	4	$1 \oplus x_1 \oplus x_2 \oplus x_2x_3$	15564	4	$x_1 \oplus x_2x_3$
13260	0	$x_1 \oplus x_3$	26214	0	$x_0 \oplus x_1$
13875	2	$1 \oplus x_1 \oplus x_3 \oplus x_0x_3 \oplus x_2x_3 \oplus x_0x_2x_3$	26265	0	$1 \oplus x_0 \oplus x_1 \oplus x_3$

Table 1. List of $(3,10)$–asynchrony immune CA rules of neighborhood size 4.

Another possible direction to explore is related to the maximum nonlinearity achievable by AI CA rules. For all even $d \in \mathbb{N}$, *bent* boolean functions $f : \mathbb{F}_2^d \to \mathbb{F}_2$ are those reaching the highest possible nonlinearity, which is $Nl(f) = 2^{d/2-1}$. Hence, an interesting question would be if it is possible to design an infinite family of bent AI CA.

A fact which could be useful for computer search of AI rules is that an infinite CA is *surjective* if and only if its finite counterpart is balanced for all lengths $n \in \mathbb{N}$, where balancedness corresponds to 0–AI. Thus, it would make sense to limit the search only to surjective CA, by adapting for instance Amoroso and Patt's algorithm [1].

Acknowledgements. The author wishes to thank the anonymous referees for their suggestions on how to improve the paper and extend the results for future research.

References

1. Amoroso, S., Patt, Y.N.: Decision Procedures for Surjectivity and Injectivity of Parallel Maps for Tessellation Structures. J. Comput. Syst. Sci. 6(5): 448-464 (1972)
2. Bertoni, G., Daemen, J., Peeters, M., Assche, G.V.: The KECCAK reference. http://keccak.noekeon.org/ (2011)
3. Carlet, C.: Boolean Functions for Cryptography and Error-Correcting Codes. In: Crama, Y., Hammer, P.L. (eds.) Boolean Models and Methods in Mathematics, Computer Science, and Engineering. Cambridge University Press, New York (2010)
4. Carlet, C.: Vectorial Boolean Functions for Cryptography. In: Crama, Y., Hammer, P.L. (eds.) Boolean Models and Methods in Mathematics, Computer Science, and Engineering. Cambridge University Press, New York (2010)
5. Chor, B., Goldreich, O., Håstad, J., Friedman, J., Rudich, S., Smolensky, R.: The Bit Extraction Problem of t-Resilient Functions. In: 26th Annual Symposium on Foundations of Computer Science, pp. 396–407. IEEE Press, New York (1985)
6. Formenti, E., Imai, K., Martin, B., Yunès, J.-B.: Advances on Random Sequence Generation by Uniform Cellular Automata. In: Calude, C.S., Freivalds, R., Kazuo, I. (eds.) Computing with New Resources. LNCS vol. 8808, pp. 56–70. Springer, Heidelberg (2014)
7. Leporati, A., Mariot, L.: Cryptographic Properties of Bipermutive Cellular Automata Rules. J. Cell. Aut. 9(5–6):437–475 (2014)
8. Martin, B.: A Walsh Exploration of Elementary CA Rules. J. Cell. Aut. 3(2):145–156 (2008)