THE PUBLISHING HOUSE
OF THE ROMANIAN ACADEMY

PROCEEDINGS OF THE ROMANIAN ACADEMY, Series A,
Volume 14, Special Issue 2013, pp. 365–372

APPLICATIONS OF NATURAL COMPUTING IN CRYPTOLOGY: NLFSR BASED ON HYBRID CELLULAR AUTOMATA WITH 5-CELL NEIGHBORHOOD

Radu DOGARU[*], Ioana DOGARU[*]

[*] University "Politehnica" of Bucharest, Dept. of Applied Electronics and Information Engineering, Natural Computing Laboratory
Corresponding author: Radu DOGARU, E-mail: radu_d@ieee.org

This work reviews some important issues in natural computing that are of interest for cryptology. In particular we focus on our recent results in defining a particular class of hybrid cellular automata (HCA) with 5 cells neighbourhood as discrete-time chaotic maps, with very good cryptographic properties. Such structures are better alternatives to other chaotic maps since they are using the hardware resources with a maximal efficiency and have no transient times associated to the convergence to the main cycle. Based on the algebraic normal form (ANF) representation of the HCA-rule it is proved that our HCA are in fact nonlinear feedback shift registers NLFSR, recently gaining increasingly interest for cryptographic applications (such as stream ciphers). It is shown that the FPGA resources are optimally allocated as a consequence of using the ANF representation of cells.

Key words: natural computing, cellular automata, chaotic dynamics, nonlinear feedback shift register, pseudo-random number generators, algebraic normal form

1. INTRODUCTION

Cryptology applications may be successfully approached by applying certain natural computing [1] techniques. Particularly, in this work we focus on a particular class of nonlinear dynamic systems inspired from nature; namely, the cellular automata. Cellular automata fit in the more general network model represented in Fig.1. At its very fundamental level (mathematical description) the network is a nonlinear dynamic system. Cells are associated with state variables (scalars or grouped in a vector structure) while connectivity links represent (nonlinear) functional relationships between states. These functional relationships include a certain number of parameters (the cell's gene in Chua's parlance [2], depicted as w_{ij} on links in Fig.1). Finding these parameters (as well as the functional expressions) such that the model fits specific goals represents the design tasks. For instance, two goals may be of interest in cryptology:

i) Designing a network capable to generate pseudo-random sequences that are difficult to decode in case of attacks (in this case genes are tuned such that in the end the network respond properly to certain batteries of tests [3]. Often the nonlinear network is designed such that it generates chaotic sequences [4][5][6] but more traditional models such as linear and nonlinear feedback shift register (NLFSR) [7][8] and cellular automata CA [9] also fit in the same category;

ii) Using partially available information from encrypted messages (as an intruder) and consider the nonlinear network as an adaptive one (e.g. using neural network paradigms) that is eventually capable to "decrypt" the structure behind a pseudo-random generator or a more sophisticated cryptographic system. This second goal is grounded on Takens's embedding theorem [10] and further methods built upon it (for instance [11]). Takens theorem allows the creation of hidden state variables in a network model based on a limited (but large enough) set of past observations. Since the accuracy of the model is better for lower number of state variables in order to avoid decryption based on Takens theorem, an extremely large state space for the cryptographic sequence generator must be considered. From this perspective the low (1-dimensional) logistic map (with no delay between transmitted samples, or other scrambling methods) is very fragile against attacks using Takens theorem [4]. On the other hand cellular automata CA were recognized as very good dynamic networks to generate cryptographic sequences since the number of their cells can be chosen as an arbitrarily large one.

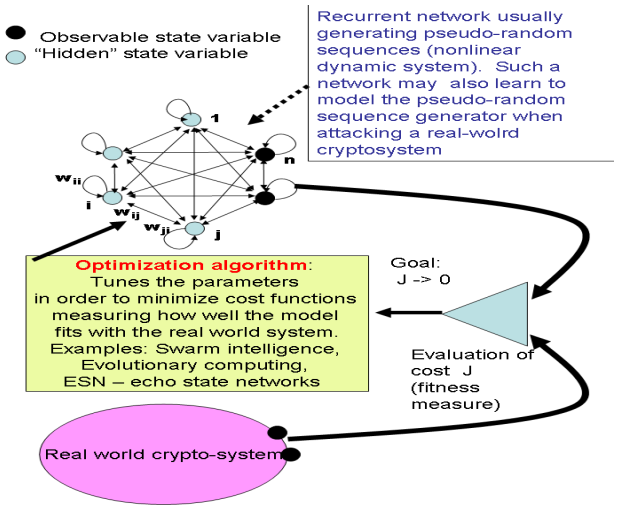

Fig. 1 – A natural computing approach to problems in cryptology: a) generating good pseudo-random sequences to be exploited in cryptosystems; b) to identify and replicate cryptographic systems (attack problems) based on consecutive observables from an encrypted transmission (real-world crypto-system).

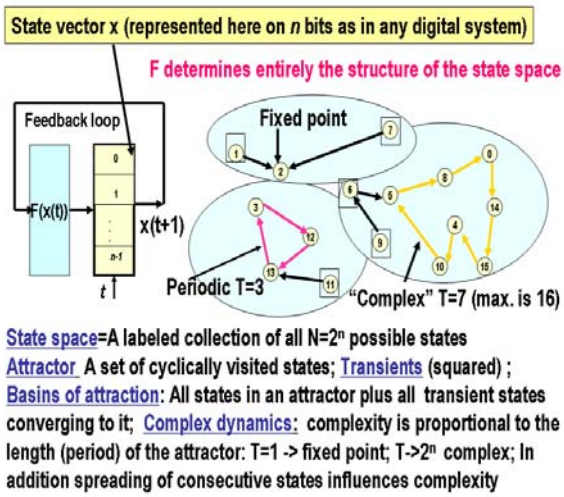

Fig. 2 – General structure of a nonlinear map in a digital implementation (discrete-time, finite precision) and the main concepts associated with it.

Mathematically, the nonlinear dynamic network is an ordinary differential equation (ODE) in the case of continuous time or a finite difference equation in the case of discrete time. In the next, we will focus only on discrete-time systems associated with their digital system implementations, ensuring reliable reproduction of identical networks as needed often at both transmitter and receiver.

In this paper, recent results on identifying a large class of cellular automata with good cryptographic properties are exposed. Such systems may be further used in various cryptosystems as pseudo-random number generators. First, in section 2, a general model for the nonlinear dynamical system is presented using discrete-time and finite precision representation of state variables. Several performance descriptors are reviewed and introduced. Particularly we will stress on conservative systems (i.e. without transients) with maximal length cycle and good randomness (an equivalent of chaotic behaviour for the case of finite state space and discrete time). Section 3 presents basic notions of cellular automata, introducing the hybrid cellular automata (HCA) with 3 and 5 cells neighbourhoods. In Section 4 a methodology is given for representing local rules in algebraic normal form (ANF) and for the identification of all nonlinear HCA (HCA NLFSR) with very good cryptology properties. Properties and conclusions are also included here.

2. DISCRETE-TIME NONLINEAR DYNAMICAL SYSTEMS AS CHAOTIC MAPS, PERFORMANCE DESCRIPTORS

In the general case, a discrete-time nonlinear dynamic system is given by equation: $\mathbf{x}(t+1)=F(\mathbf{x}(t))$, where $\mathbf{x}$ is a *state variable.* The above is called a *map* and if the dynamic behaviour is chaotic the map is said to be *chaotic.* In recent year chaotic maps gained increased interest for applications in cryptology [5][6]. Quite often they are used to build cryptosystems where the underlying dynamical system is used as pseudo-random number generator. A widely known example of a chaotic map is the one dimensional logistic map [12][13] where $F=\lambda x(1-x)$. From a practical perspective (digital implementation, where all state variables can be associated with a unique register with n bits) any nonlinear map can be associated with an automaton-type as presented in Fig. 2. For instance, if the unique x state variable of the logistic map is implemented in a 32-bits floating point representation the associated register in Fig. 2 has $n=32$ bits and the nonlinear function $\mathbf{F}$ is implemented as a combinational digital circuitry. Often, in digital implementations fixed-point representations of the state variables are preferred since they ensure a better efficiency (less occupied

resources). In [14] an interesting comparative study of various chaotic maps using various sizes *n* with fixed or floating point implementations in FPGA is given. In [15] we show that the general structure of a digital implementation for a nonlinear map in Fig.2 includes cellular automata (CA) as particular case. Moreover, the CA have several advantages over nonlinear maps, such as the lack of transients (for certain kind of cellular automata called *conservative* CA), easiness of scaling to high dimensionality *n* (with direct implications for decreasing the probability of successful attack), better efficiency in hardware (*e.g.* FPGA), to name just a few. More details about cellular automata as chaos generators and their design are given in Section 3. In the following we introduce the main performance descriptors and discuss them from the cryptology perspective. As a general rule, in cryptology we are interested to design systems with very long cycle length *L* (as close as possible to the maximal length i.e. $N = 2^n$), large state dimension *n*, and a "chaotic" character of the longest cycle, no transients, and an efficient and scalable VLSI implementation:

Transients: They are also called "ephemeral states" (Fig.2) since they are never visited twice during the dynamics of the system. Most chaotic maps reported so far have such transients and, to our knowledge there is no consistent theory to predict the transient length for a given system (although some useful computational evaluations were done in [13]). It is also clear that the initial state influences the transient length (i.e. the number of iterations until entering a *cycle*). For cryptographic application is useful to avoid the existence of transients since they are not useful states and consequently their presence will diminish the "chaotic" cycle length ($L<N$) also requiring some lost cycles until entering the useful cycle. Recently [16] within the cellular automata framework, *conservative* CA was defined (it also apply to any network model in Fig.2) as those having the property that *no state is a transient or ephemeral state*. Both LFSR and NLFSR (linear feedback shift register) used for long as pseudo-random sequence generators have this property as well. The cellular automata in this paper are also conservative and consequently they have no transients.

Cycle length (L): As seen in Fig.2 for different feedback functions **F** (parameters of **F** often represent the encryption key) different state space profiles are obtained. They consist of cycles of various lengths. Ideally, for cryptographic applications, the state space profile should be represented by a single cycle with the maximal length $L = N = 2^n$. In practice solutions where one very large cycle with $L \cong N$ coexists with several very small cycles are acceptable. The closest is L/N to 1 the better. For practical reasons it is also convenient that a particular state (such as "all bits in 0") is enclosed in the dominant cycle (the one with maximal length).

Randomness (degree of chaos): A very long cycle is not necessarily a random one. A good counter-example is the counting automaton (known as "counter" in digital circuit parlance). It has a maximal cycle length $L=N$ but the transition from one state to the consecutive one is rather smooth, often only one bit is changing. Moreover, altering one bit in a counting state gives no dramatic influence on the trajectory. According to the classical chaos theory, a small change in the state variable must influence dramatically the values of the consecutive states (many different bits in our case). Since here we discuss about finite state space, the use of Lyapunov exponents (defined in the context of continuous state variables) is not suitable. Instead, in [17] we introduced a randomness measure that may be conveniently computed. It characterizes very well the randomness of any cycle. We are in particular interested by the randomness of the *dominant cycle*. The measure of randomness was defined observing that in a "chaotic" automata the average Hamming distance between consecutive binary vector states (as given by the *n* cell outputs) becomes *n/2* instead of 1 for counters. Therefore, for any arbitrary cycle C_j of length L_j a *scattering coefficient* S_j is defined by averaging the Hamming distances between all consecutive binary vector states in that cycle:

$$S_j = \frac{1}{nL_j} \sum_{k=1}^{L_j} \sum_{i=1}^{n} \left| x_i(k) - x_i(k-1) \right| \tag{1}$$

where *k* is the time index of consecutive states in the cycle *j*. A *degree of chaos* λ_j is defined such that it becomes maximum if $S_j = 0.5$ and zero for the extreme, non-chaotic cases of both fixed points and period 2 cycles (with $S_j = 0$ and $S_j = 1$ respectively):

$$\lambda_j = 1 - \left| 2S_j - 1 \right| \tag{2}$$

The *degree of chaos* may be regarded as qualitatively similar to the Lyapunov exponent used in continuous-state systems to characterize chaotic behaviours. In our case its largest value is $\lambda_j = 1$ indicates the highest degree of randomness in a finite-length cycle of an automata network.

Efficiency of hardware utilization: Such a descriptor indicates: i) how many basic devices are needed to implement the feedback function F for a given *n* and ii) how well the state space is used (*i.e.* the ratio *L/N*). As shown in [15] usual chaotic maps (logistic in that case) would need far more hardware resources than hybrid cellular automata introduced first in [18]. For instance, according to [15] in the case *n=32* , 1317 LUTs (basic logic cells in FPGA) are needed while only 32 LUTs (in general *n* LUTs) are needed for the HCA101 cellular automata. On the other hand, the HCA101 has a far better utilization of the state space, for example, as shown in [15] in the case *n=21*, $L = 2097151 = 2^{21} - 1 = N - 1$ for HCA with ID=101 *i.e.* all states except 1 are included in the dominant chaotic cycle (also there are no transients) while for the same size *n* of the register the logistic map with $\lambda = 3.7$ (proved to be chaotic in floating point implementations) has the length of the dominant cycle only $L = 883$ (*i.e.* a fraction of only 0.0004 of all states), all other states belonging to transients! Such results indicate why it is desirable to consider a special case of cellular automata, the hybrid cellular automata (HCA) instead of (classical) chaotic maps. They are conservative maps and have the highest degree of randomness dominant cycle.

3. HYBRID CELLULAR AUTOMATA

Cellular automata have a long history of their use in cryptology. Many patents dealing with cellular-automata based cryptographic systems are issued, (first patent on this issue [19]) and their study is the focus of many research papers (*e.g.* [9][20][21]). The cellular automata model fits the general model of a nonlinear map implemented as a digital system, each bit of the register being now associated to one cell and the feedback function **F** is now defined as a collection of *n* similar local (Boolean) functions acting in a *neighbourhood* of *m* cells. The reason why cellular automata are so popular in cryptology is the easiness to scale them to arbitrarily large *n* (as discussed above, a large size of the state space makes cryptographic attacks more difficult) due to the locality of the **F** mapping. Among the first cellular automata types that were widely investigated are the elementary cellular automata (ECA) [Wolfram 1983] where the neighbourhood size is *m=3*. Randomness was then associated with the evolution of cellular automata with rule (ID) 30, also adopted as random number generator in Wolfram's "Mathematica". Consequently, Chua [16] did an exhaustive research on all possible 256 ECA and introduced the concept of *conservative* cellular automata. As shown, the ECA with ID=30 is not a conservative cellular automata, consequently it has the transients. In [16] it is shown that CA with odd number *n* of cells governed by rules ID = 45 (and its other 3 equivalents ID = 75,89, and 101) and the complemented outputs ID = 154 (or its equivalents, ID = 166, 180, 210) are the only *non-linear rules* leading to *conservative* dynamics with a high degree of complexity. Independently, in [17] we were able to show that in addition to this property, the cellular automata with ID = 45,75,89,101 have very good randomness properties, although the lengths of the dominant cycle are not maximal for any *n*. In addition these automata networks allow synchronization between a transmitter and a receiver with identical structure using only 1 bit (binary synchronization), a feature that may be conveniently exploited particularly in communication systems. The good randomness and cryptographic properties of CA with ID=101 and it equivalents is confirmed in [9] using standard batteries of statistical tests [3]. Later, in [15] we introduced the concept of a hybrid cellular automata (HCA) demonstrating that for a proper choice of a mask vector of size *n* (complementing the outputs of some ID = 101 cells) one can maximize *L* such that it will reach a value very close to *N* (thus, improving the cryptographic properties of the random number generator [30]). Several applications were proposed [22][23], particularly interesting is the one in [23] where such a HCA is used as "chaotic counter" to scan a video sensor leading to a very compact yet versatile system which ensures compression, encryption and spectrum distribution at the same time and with very little hardware requirements. Such solutions are very convenient for low power remote sensing applications (*e.g.* surveillance etc.). As seen, instead of various parameters of the chaotic maps (*e.g.* the λ parameter) in the case of CA or HCA maps the rule (also called ID, to be detailed next) takes the role of parameter, being associated with part of the key space. The masks in the HCA may represent part of the encryption key as well as the initial state. In order to increase the key space one needs to consider larger neighbourhoods such

that more IDs will be identified as useful in terms of good cryptographic properties. Consequently, in Section 4 we discuss the case *m=5* (5 cells neighbourhood) and locate many other ID (from a huge space of 2^{32} – about 4 billion) to generate HCA with good cryptographic properties. Consequently the key space is considerably expanded than in the case of *m=3* neighbourhood.

3.1. HCA Structure

To explain the HCA structure we consider first the case *m*=3 (3 cells neighbourhood). It expands naturally to a 5-cell neighbourhood. Fig. 3 presents two *m=3* HCA automata structures that were successfully tested for having both good cryptographic (as exposed in section 2) and binary synchronization properties. Note that they can be operated in either autonomous mode (as is the case in the transmitter system Tx) or with one input forced by the synchronization signal (as is the case in the receiving system Rx).

The discrete-time dynamics of the hybrid cellular automata (HCA) in Fig. 3 is given by the next equation, which applies synchronously to all *n* cells (a cell is identified by an index $i \in \{1,2,..n\}$):

$$x_i^T(t+1) = m_i \oplus Cell\left(x_{i-1}^T(t), x_i^T(t), x_{i+1}^T(t)), ID\right) \tag{3}$$

where the upper index "*T*" stands for the transmitting CA counter, $\oplus$ is the logical XOR operator and $Cell(u1,u2,u3,ID)$ is a Boolean function with 3 binary inputs (*u1,u2, and u3*), also called the CA (local) rule. A periodic boundary condition is also assumed *i.e.* the leftmost cell (*i=1*) is connected to the rightmost one (*i=n*). The binary *mask vector* $\mathbf{m} = [m_1, m_2, .., m_n]$ can be optimized [18] (so far our programs and limited computational time allow up to $n \le 29$) to obtain a maximal cycle length ($r = L/2^n \to 1$). Since there are many near-optimal length masks, the mask itself can be considered as part of the encryption key. There are many possibilities to increase the key space. The one described in detail in the next section assumes a larger neighbourhood and consequently many types of cells ensuring the desired properties. One possibility, recently investigated, is to consider an alteration of the regular cellular topology into a "small worlds" one, as shown in Fig. 3 (right side). Essentially the model is described by the same equation (3) where the mask can be removed (*i.e.* all $m_i = 0$) but where the optimization of the maximal cycle length may now be achieved using a random search process in a space of permutations (one or more pairs of outputs are swapped as shown in Fig.3). The mask is now replaced by the set of swapping pairs (i_{sw}, j_{sw}). The above equation (3) extends easily to larger neighbourhoods such as *m*=5 considered herein by adding 2 additional inputs to the cell, located on the rightmost and leftmost positions (*i-2,* and *i+2*). For any neighbourhood the relationship between inputs and the output local CA rule can be characterized in two different ways and conversion functions are available via [24]: a) ***Truth-Table*** *(TT)* representation: This is the most widely used representation. The rule is characterized by a binary vector $Y = [y_{N-1}, y_{N-2}, ..., y_0]$. Its representation in decimal basis is called a rule identifier (ID). The output y_k is a binary number assigned to the cell's output when its inputs ordered as a binary vector $[u_n, u_{n-1}, .., u_1]$ are the binary representation of *k;* b) ***Algebraic Normal Form (ANF):*** This form is described by a binary vector $C = [c_0, c_1, ...c_N]$ (using the method in [24] a unique conversion from **Y** to **C** and vice-versa exists) such that its coefficients are multipliers of an algebraic representation on the GF_2 exemplified next for the case of *m=3* neighborhood:

$$y = c_0 \oplus c_1u_1 \oplus c_2u_2 \oplus c_3u_2u_1k_3u_3 \oplus c_4u_3 \oplus c_5u_3u_1 \oplus c_6u_3u_2 \oplus c_7u_3u_2u_1 \tag{4}$$

Note that in general (for any size *m* of the neighbourhood) c_k is the multiplier of a product (logical AND) of all input variables in a binary vector $[u_n, u_{n-1}, .., u_1]$ corresponding to 1 in the associated binary vector representing *k.* For example, in the case of *m=3* (above equation) for $k = 5 = 101_2$ only the inputs corresponding to 1 in the input string $u_3u_2u_1$ are selected to be multiplied resulting in the term $c_5u_3u_1$.

The ANF representation is extremely useful for FPGA implementations since it allows a direct translation into a simple VHDL line describing the entire HCA structure [25]. The resulting synthesized structure has a very efficient use of the resources, better than reported in other works for similar automata [26]. On the other hand, ANF is very useful to reveal whether the automata network is a linear one (*e.g.* in the same category with LFSR and other hybrid cellular automata that are mostly reported so far, typically

with *m=3* and rules ID=90, ID=150 [27]) or a nonlinear one (from the same category with NLFSR, recently gaining a lot of interest [7][8][28] as having a better immunity to attacks than LFSR). A nonlinear automaton has at least one term with more than 2 inputs in the ANF equation (4).

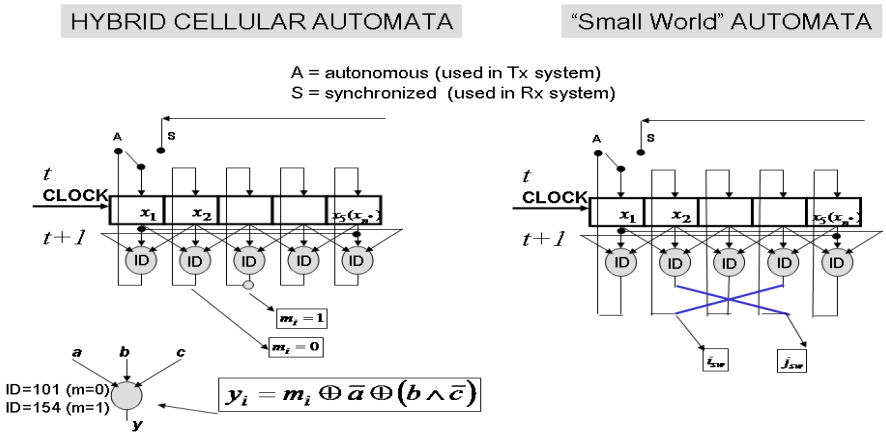

Fig. 3 – Two structures of HCA with *m=3* neighbourhood: a) left – the "standard" HCA, b) the "small-worlds" HCA where some of the outputs are swapped. Such chaotic counters may be operated in either autonomous or synchronized mode.

4. RESULTS AND CONCLUSIONS

In order to investigate the very large family of 5-cells neighbourhood automata (there are $2^{2^5} = 4294967296$ different cells possible) we adopted the following strategy:

i) a set of software tools was developed as follows: a) Matlab tools for conversions between ANF and TT representations; b) a Matlab tool to establish the attractor profile revealing the cycles, their lengths and their associated randomness for an arbitrary number of cells *n* (Fig. 4). This program is used during a search process to locate ID-s with potential for good cryptographic properties [3][30]; c) A compiled C program to optimize the mask in order to get a very long cycle (state "0" corresponding to all bits equal to 0 is included in the longest cycle) and evaluate the average number of cycles for binary synchronization (Figure 5). Examples are given for ID=1347465135 corresponding to the ANF: $y = 1 \oplus u_5 \oplus u_1 \oplus u_5 u_1$.

Note than finding the optimal mask is a computationally intensive processes involving the running of long cycles for several thousands of trials. The computational complexity is $O(2^n)$ and it takes about 3 minutes for *n*=17 and 4 times more for *n*=19.

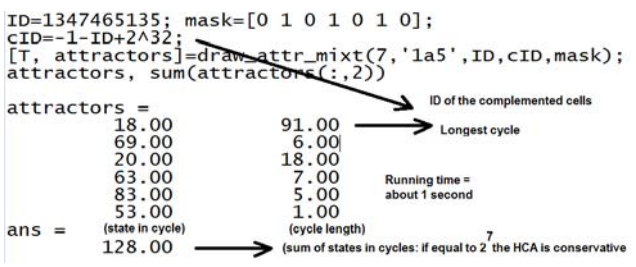

Fig. 4 – Detecting the attractor structures for a given ID and mask.

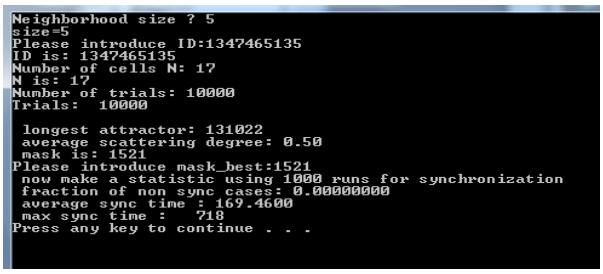

Fig. 5 – Optimization of mask.

ii) We considered the ANF representation to describe all possible families of 256 HCA corresponding to 3 inputs in the 5-cell neighbourhoods and use the conversion software to calculate ID and then evaluate the profile of attractors keeping only the desired CA. The situations that are symmetrical with respect to the central cell were not considered because they can be easily recovered by simply replacing $u_k = u_{6-k}$. A synthesis of the results is given in the following table.

Distribution of inputs in the neighbourhood	ID	ANF
[u5,-,u3,-,u1] 8 HCAs with good cryptographic properties	2947502160 2779097770 4126476810 2863290970 + 4 complemented versions of the above	$y = 0 \oplus u_3 \oplus u_1 \oplus u_5u_1$ $y = 0 \oplus u_5 \oplus u_1 \oplus u_3u_1$ $y = 0 \oplus u_5 \oplus u_1 \oplus u_5u_3$ $y = 0 \oplus u_5 \oplus u_3 \oplus u_3u_1$
[u5,u4,-,u2,-] and [u5,-,u3,u2,-]	NONE	-
[-,-,u3,u2,u1]	4043247360 4027576560 4279173360 4042264560 + 4 complemented versions of the above	$y = 0 \oplus u_2 \oplus u_1 \oplus u_3u_2$ $y = 0 \oplus u_3 \oplus u_1 \oplus u_2u_1$ $y = 0 \oplus u_3 \oplus u_1 \oplus u_3u_2$ $y = 0 \oplus u_3 \oplus u_2 \oplus u_2u_1$
[-,u4,u3,u2,-] equivalent to [u3,u2,u1] (3 cells neighborhood)	8 Boolean functions already described in [Dogaru HCA]	The above formulare where u_k is replaced with u_{k+1}

At a closer inspection of the ANF formulae in the above table a very interesting conclusion follows: In all cases the same logical functions with 4 inputs is performed namely: $y = m_i \oplus u_a \oplus u_b \oplus u_cu_a$ where m_i is bit *i* of the mask and all 6 possible permutations of the indices *a,b,c* are possible. Further studies indicates that the relationship between *a,b* and *c* leading to good cryptographic HCA is: $a - b = b - c = h$ where *h* is an integer. For instance, if $h = 2$ $(a,b,c) = (5,3,1)$ and all 6 permutations leading to the IDs from the first line of the above table. Higher values of *h* will make sense for neighbourhoods larger than 5. For instance, h = 3 will lead to $(a,b,c) = (7,4,1)$ corresponding to *m=7* cell neighbourhood. For any of these possible combinations one may optimize masks as seen in Fig. 5. For instance ID=1347465135 has the best mask found so far 19801 leading to *L=N-1=131071.* In this case as for many other masks of the large family of CA discussed above, the randomness coefficient is maximal, *i.e.* $\lambda = 1$. A detailed analysis using standard batteries of statistical tests reveals that all HCA from the family generated by the unique logical functions with 4 variables have very good cryptographic properties [29] expanding similar results reported by other authors [9] for the elementary CA with ID=101 which also uses a form of the above logical function as local rule.

Concluding, in this work we approach the design of good random number generators using natural computing concepts, mainly the one of cellular automata. It is shown that all major paradigms for cryptographic generators (chaos based, linear and nonlinear shift register and cellular automata) fit in the same unique model of a nonlinear network with a binary state vector represented on *n* bits. It is shown that comparing with chaotic maps the use of hybrid cellular automata as nonlinear maps brings the advantage of an efficient use of the state space (no transients). Moreover, the case of 5-cell neighbourhood (using cells with 4 inputs, such that each HCA cell will correspond to 1 basic computational element (LUT or LE) in FPGA technology as already was demonstrated in [25]. The algebraic normal form was conveniently used for synthesis since equation (4) translates directly into one VHDL line ensuring the description of the underlying CA. Further research will focus on identifying all cryptographically useful HCA-NLFSR with 4 and 5 inputs rule in a 5-cell neighbourhood. For such cells, a VHDL description is already available which shows after synthesis that 2 LUTs are needed for each HCA cell (doubling the necessary hardware resources for the same *n* when compared to the case of 3-inputs from the 5-cell neighbourhood). Preliminary results show that among cells with 4-inputs, the HCA with ID=3432828060 was found as having similar cryptographic properties with the HCA family discussed above, but in addition it has this properties for an arbitrary *n* number of cells (the other HCAs do not have the "no transients" property for even *n.* Such a property is particularly useful in applications such as [23] in addressing square image sensor that will require an even number *n* of bits. Another further open question is to develop a systematic theory for finding the optimal masks analytically and to explain why only a very small number (among the many possible IDs) are both conservative and also have good randomness properties. Such problems are recognized as open problems in the NLFSR research community as well [29].

REFERENCES

1. Leandro N. De Castro, *Fundamentals of Natural Computing: Basic Concepts, Algorithms, and Applications*, Chapman & Hall/CRC Computer, 2006.
2. L. O. Chua, *CNN: a Vision of Complexity*, International Journal of Bifurcation and Chaos, Vol.7, No.10, pp. 2219–2425, 1997.
3. National Institute of Standards and Technology, Federal Information Processing Standards Publication 140-2: *Security Requirements for Cryptographic Modules*, US Government Printing Office, Washington, 1999.
4. G. Alvarez, S. Li, *Some basic cryptographic requirements for chaos-based cryptosystems*, Int. J. Bifurcation Chaos Appl. Sci. Eng. 16, pp. 2129–2151, (2006).
5. N.K. Pareek, Vinod Patidar, K.K. Sud, *Cryptography using multiple one di-mensional chaotic maps*, Commun. Nonlinear Sci. Numer. Simul. 10 (7) (2005), pp. 715–723.
6. W. M. Tam, F. C. M. Lau, and C. K. Tse, Digital Communications With Chaos. Oxford, U.K.: Elsevier, 2007.
7. Tomasz Rachwalik, Janusz Szmidt, Robert Wicik, and Janusz Zablocki, *A Generation of Nonlinear Feedback Shift Registers with special-purpose hardware*, Cryptology ePrint Archive, Report 2012/314, June 2012, http://eprint.iacr.org/2012/314
8. E. Dubrova, *How to speed-up your NLFSR-based stream cipher*, in Proceedings of Design, Automation & Test in Europe Conference & Exhibition (DATE '09), pp. 878–881, 2009.
9. Franciszek Seredynski, Pascal Bouvry, Albert Y. Zomaya, Cellular automata computations and secret key cryptography. Parallel Computing 30(5-6): 753–766 (2004)
10. F. Takens, *Detecting strange attractors in turbulence*, in D. A. Rand and L.-S. Young. Dynamical Systems and Turbulence, Lecture Notes in Mathematics, vol. 898. Springer-Verlag. pp. 366–381, 1981.
11. M.B. Kennel, H.D.I. Abarbanel, *False neighbors and false strands: A reliable minimum embedding dimension algorithm*, in Phys. Rev. E, VOl. 66, 026209 (2002) [18 pages].
12. A. Kanso, N. Smaoui, *Logistic chaotic maps for binary numbers generations*, in Chaos, Solitons and Fractals, 40 (2009), pp. 2557–2568.
13. Adriana Vlad, A. Luca and M. Frunzete, *Computational Measurements of the Transient Time and of the Sampling Distance That Enables Statistical Inde-pendence in the Logistic Map*, Lecture Notes in Computer Science, 2009, Volume 5593/2009, 703–718.
14. P. Giard, G. Kaddoum, F. Gagnon, C. Thibeault, *FPGA implementation and evaluation of discrete-time chaotic generators circuits*, In proceeding of: IECON 2012 - 38th Annual Conference on IEEE Industrial Electronics Society, pp.3221–3224, 2012
15. R. Dogaru, *HCA101: A chaotic map based on cellular automata with binary synchronization properties*, in Proceedings of The 8'th Int'l Conference on Communications-COMM2010, 10-12 June, Bucharest, Romania, pp. 41–44.
16. L.O. Chua, A Nonlinear Dynamics Perspective of Wolfram's New Kind of Science (Vol I-IV), World Scientific Series on Nonlinear Science, Series A - Vol. 57, 68, 76, World Scientic Publishing Company, 2006, 2009, 2011.
17. R. Dogaru, I. Dogaru, and H. Kim, *Binary chaos synchronization in elementary cellular automata*, Int. J. Bifurcation Chaos,Volume: 19, Issue: 9, pp. 2871–2884, 2009.
18. R. Dogaru, *Hybrid Cellular Automata as Pseudo-Random Number Generators with Binary Synchronization Property*, in Proceedings of the International Symposium on Signals Circuits and Systems, Iasi Romania, July 2009, pp. 389-392.
19. S. Wolfram, *Random sequence generators*, US patent 4691291A, 1987.
20. D. Das, *A Survey on Cellular Automata and Its Applications*, Global Trends in Computing and Communication Systems, Communications in Computer and Information Science Volume 269, 2012, pp 753–762.
21. R. Dogaru, *Systematic design for emergence in cellular nonlinear networks with applications in natural computing and signal processing*, Springer-Verlag, Berlin Heidelberg, 2008.
22. R. Dogaru, H. Kim, I. Dogaru, *Binary Synchronization in Cellular Automata for Building Compact CDMA Systems*, in Proceedings of the International Sym-posium on Signals Circuits and Systems (ISSCS'09), Iasi Romania, July 2009, pp. 393–396.
23. R. Dogaru, I. Dogaru, H. Kim, *Chaotic Scan: A Low Complexity Video Transmission System for Efficiently Sending Relevant Image Features*, IEEE Trans. on Circuits and Systems for Video Technology, Vol.20, Issue 2, pp. 317–321, 2010.
24. S. Ronjom, M. Abdelraheem and L. E. Danielsen, *TT and ANF Representations of Boolean functions*, in Online Database of Boolean Functions, 2007. Available: http://www.selmer.uib.no/odbf/help/ttanf.pdf
25. I. Dogaru and R. Dogaru, *Algebraic Normal Form for Rapid Prototyping of Elementary Hybrid Cellular Automata in FPGA*, in Proceedings ISEEE 2010 (September 2010, Galati, Romania), pp. 273–276.
26. P. Anghelescu, E. Sofron, S. Ionita, L. Ionescu, *FPGA implementations of cellular automata for Pseudo-random number generation*, International Semiconductor Conference, CAS, 2006, Vol. 2, pp. 371–374.
27. K Cattell and J.Cmuzio. *Synthesis of one-dimensional linear hybrid cellular automata*, IEEE Trans. on Computer-aided design of integrated circuits and systems, 15(3):325–335, 1996.
28. M. S. Turan, *On the nonlinearity properties of maximum-length NFSR feedbacks*, Cryptology ePrint Archive, 2012/112.
29. E. Dubrova, *A Scalable Method for Constructing Galois NLFSRs With Period 2^n-1 Using Cross-Join Pairs*, in IEEE Transactions on Information Theory, (Volume:59, Issue: 1), pp. 703-709, January 2013.
30. Andrei Petrus, (coordinator R. Dogaru), *Random number generator based on cellular automata (FPGA implementation)*, Master Thesis (Romanian language), september 2012, University "Politehnica" of Bucharest.

Received May 20, 2013