2010 International Symposium on Nonlinear Theory and its Applications
NOLTA2010, Krakow, Poland, September 5-8, 2010

Uncertainty Profiles for Predicting Complex Nonlinear Dynamics in Cellular Automata: The Case of Five Cells Neighborhood

Radu Dogaru† and Ioana Dogaru†
†Dept. of Applied Electronics and Information Engineering, "Politehnica" University of Bucharest
Bvd. Iuliu Maniu 1-3, Bucharest, Romania
Email: radu_d@ieee.org, ioana_d70@yahoo.com

Abstract– Uncertainty profiles are behavioral indexes that can be easily determined for a given CA cell structure and neighborhood, as a nonlinear relationship between the probability of a cell to be in state "1" when the probabilities of the neighbor cells in the previous iterations are known. They allow to predict the nature of CA dynamics without the need to effectively simulate the dynamics. This paper reviews the method of uncertainty profile as a method for "designing for emergence" and exemplifies it first time for CAs with 5 cells neighborhoods. The predictive power of this method is demonstrated in conjunction with other recent methods for nonlinear analysis and classification of CA dynamics.

1. Introduction

Cellular automata (CA) are a subclass of nonlinear dynamic systems endowed with certain characteristics that makes them attractive as naturally inspired, massively parallel computing architectures. Their applications range from modeling to cryptography and signal processing. Unlike traditional computers, where sets of instructions are used to build algorithms tailored to certain applications, in "programming" the CA the goal is to establish the values of a relatively small set of parameters called a cell's *gene* [1] such that a certain kind of dynamic behavior, with applicative potential, will emerge in an array of identical cells defined by the unique *gene*. Such dynamics is often called *emergent* although a precise definition of emergence is lacking for the moment. In the case of Boolean CA the *gene* is a vector of binary elements $\mathbf{Y} = [y_{N-1}, y_{N-2}, ...y_0]$ defining the local Boolean function of a cell with $n = \log_2(N)$ inputs. The decimal representation of the above string will be called an ID. It is assumed that the desired dynamics is often vaguely specified (e.g. "chaotic behavior" or "must have a Class IV behavior in the sense of Wolfram", etc.). Such problem is rather general and applies for any kind of nonlinear dynamic system. Although for some dynamic behaviors (typically the equilibrium behaviors) analytic approaches are possible, such as those given by various stability theorems, the "design for emergence" is still an open issue. In some sense, defining the "programming" of nonlinear dynamic systems to perform useful functions may have the same relevance as the defining of various programming techniques which sprung the explosion of the information technology using "classic" computational structures such as the microprocessor. Advanced nano and molecular technologies are ideal for the realization of CA computational mediums that will need adequate "programming" techniques to spread a new wave of computing applications. Only recently, starting with pioneering works in the area of *local activity* [2][3] the importance of such *design for emergence* techniques was recognized. A review of the such techniques is given in [4]. In the same work the uncertainty profile method (UPM) was first exposed. It allows to establish a mathematical relationship between the *gene space* and a *behavior space* without a need to simulate the dynamical system. Its effectiveness in predicting various emergent behaviors was demonstrated for elementary CA [5][9]. Certain novel relevant dynamic behaviors with interesting applications, such as binary synchronization of chaos [6][7][8] were also put in a direct relationship to the UPM method. In [9] we investigated the relationship between our UPM method and a set of nonlinear dynamic methods for classifying elementary CA. They are assigned precisely into 6 classes as described in [10][11] from a series of monumental works dedicated to a nonlinear dynamics approach to CA. It turned out that UPM may predict quite well if a CA with a given gene is in one ore another of the 6 categories although its probabilistic nature gives also some little misclassification errors [9]. In this paper we extend the application of this method to one-dimensional CA with 5 cells neighborhoods. In Section 2 we will briefly review main concepts in designing for emergence when applied to cellular automata. Section 3 gives the definition of the uncertainty profile and formulae for its calculation in the case of our interest. Although the search space in this case is rather big (there are 2^{32}, i.e. about 4 billion possible genes) in Section 4 we demonstrate how easily one can pick genes leading to desired dynamic behaviors from the 6 classes, while UPM defines an inverse relationship between the desired behavior space and the solution gene space.

2. Cellular Automata and Design for Emergence

The nonlinear dynamic systems investigated herein are CA with *M* cells given by the following discrete time state equation which applies synchronously to all cells (a cell is identified by a spatial index $i = 1,..M$) and with a specified initial condition for all CA cells:

$$x_i(t+1) = Cell\left(x_{i-2}(t), x_{i-1}(t), x_i(t), x_{i+1}(t), x_{i+2}(t)\right), ID) \quad (1)$$

where $y_j = Cell(u1, u2, u3, u4, u5, ID)$ is the local rule or transition function of a cell given a neighborhood of *n=5* cells. A periodic boundary condition is assumed, such that when *i=1* (leftmost cell) its neighbors to the left are given by indexes *i=M, M-1*. The local function is completely specified by the gene vector mentioned in Introduction, or its associated ID. In other words, the *j* index in the above is the decimal representation of the binary word $[u5, u4, u3, u2, u1]$.

In [10][11] a precise classification and some expected behavioral properties of CA was given for all 88 basic elementary CA rules. Briefly speaking, the main feature to discriminate among classes was the length (period) of the most likely attractor (given a random initial state) and its dependency on the number *M* of CA cells. Classes 1,2,3 (with 26, 13 and 1 members) are similar in that for all of them there is a *constant* (1,2, or 3) period of the main attractor. Class 4, (or Bernoulli-shift), with 30 members, includes all CA rules leading, as for the previous ones, to a predictable period (given the gene and without simulating the CA) that depends *linearly* by the number of cells *M*. Finally, classes 5 and 6 (with 10 and 8 members respectively) include the most complex CA rules, where a prediction of the attractor length and its dependence on *M* is not possible without effectively running the CA. Rules in Class 5 are symmetric (bilateral) while those in Class 6 are not. **Quite notably, CA rules found so far by other researchers as possessing interesting properties fall in Chua's Class 6.**

As seen above, while various *behavioral descriptors* (e.g. the attractor length in the above, or Lyapunov exponents, or various entropies etc.) may be defined and associated with a *behavioral space* **B**, its association with the *gene space* **G** can be usually done only via an algorithm to simulate the nonlinear dynamics and compute the descriptors. Thus, associating a point $\mathbf{g}_1$ from the gene space to a behavior point $\mathbf{b}_1 = A(\mathbf{g}_1)$ in B is straightforward. Such an approach makes extremely difficult to design for emergence (i.e. solve the inverse problem of finding $\mathbf{g}_1$ when $\mathbf{b}_1$ is given). The difficulty stands in the lack of mathematical instruments to show the existence and to compute the inverse A^{-1} of the algorithm simulating the nonlinear dynamics. Yet, as seen in the case of stability theory, local activity theory, or as applied in the case of Classes 1-4 above, using nonlinear theory tools makes possible to directly locate genes for given behaviors. Still, remains of interest to locate genes associated with complex behaviors such as those in Classes 5-6 that may be also subdivided into even finer categories (e.g. among chaotic behaviors one may identify synchronizable behaviors etc.). To achieve this goal we introduce next uncertainty profile as a vector of behavioral indexes. Associations between behaviors described previously in various classes and concrete values of uncertainty profiles were established in [9] and may be used to predict the behavior for arbitrary neighborhoods and sizes. In the next we will exemplify for the case of *n=5* cells in a neighborhood (or, using a taxonomy introduced in [4] CA belonging to '1a5' family).

3. Uncertainty Profiles as Behavioral Descriptors

Uncertainty profiles are behavioral indexes that can be easily determined for a given CA cell structure and neighborhood. The idea is to consider that all initial state (*t*=0) cells are in a quiescent state $q \in \{0,1\}$ with probability $p_k = 1$, except a group of *n* neighbor cells assigned state 1 with probability 0.5 or maximum *uncertainty* $u_k = 1$. In our theory the uncertainty of a cell "*k*" is computed as $u_k = 1 - |2p_k - 1|$ and consequently it is 0 in either cases when a cell is an "sure" state (0 or 1). As seen in Fig.1 the effective value of u_k is less important than its spatial spread after one CA iteration (*t=1*), which relates to various types of CA behaviors as shown in [4].

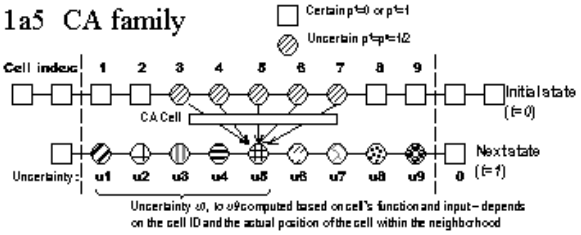

Fig.1. Uncertainty profile and its spread for CA with 5 cells in the neighborhood.

For a given local Boolean function and probability of its inputs, it is possible to compute the output probability using an information theoretic approach. Details are given in [4], resulting that the output probability is a degree *n* polynomial with respect to the input probabilities and with $N = 2^n$ coefficients given by weighted summations of the gene's **Y** bits. When the neighborhood arrangement is given, as shown in Fig.1, the polynomial output probability formulae allows to compute the uncertainty for all 2*n-1*=9 positions that are affected in the next state, i.e. resulting in an uncertainty vector $\mathbf{U} = [U_1, U_2, .., U_9]$. Without entering into details (explained elsewhere [12]) it turns out that for input uncertainties equal to 1 in the *n=5* additional cells (initial state) two *representative probability matrices* $\mathbf{R_0}$ and $\mathbf{R_1}$ can be precisely defined, each having *2n-1* lines and $N = 2^n$ columns. R_0 is given below while R_1 is its mirrored version (last column of R_0 becomes the first of R_1, etc.):

$$\mathbf{R}_0 = \frac{1}{32}\begin{bmatrix}
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&16&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&16\\
0&0&0&0&0&0&0&8&0&0&0&0&0&0&0&8&0&0&0&0&0&0&0&8&0&0&0&0&0&0&0&8\\
0&0&0&4&0&0&0&4&0&0&0&4&0&0&0&4&0&0&0&4&0&0&0&4&0&0&0&4&0&0&0&4\\
0&2&0&2&0&2&0&2&0&2&0&2&0&2&0&2&0&2&0&2&0&2&0&2&0&2&0&2&0&2&0&2\\
1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&2&2&2&2&2&2&2&2&2&2&2&2&2&2&2&2\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&4&4&4&4&4&4&4&4&4\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&8&8&8&8&8\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&16&16
\end{bmatrix}$$

Consequently the relationship connecting the gene space (represented by vectors **Y** or their associated ID) with the behavior space (represented by **U** vectors) is given by the next equations:

$$\mathbf{P_0} = \mathbf{R}_0\mathbf{Y} \text{ and } \mathbf{P}_1 = \mathbf{R}_1\mathbf{Y} \quad (2)$$

$$\mathbf{U}_0 = 1 - |2\mathbf{P_0} - 1| \text{ and } \mathbf{U}_1 = 1 - |2\mathbf{P}_1 - 1| \quad (3)$$

$$\mathbf{U} = 0.5[(2 - y_o - y_o y_{N-1})\mathbf{U}_0 + (y_o + y_o y_{N-1})\mathbf{U}_1] \quad (4)$$

The above relationships hold for any neighborhood provided that their corresponding R matrices are predetermined. It is clear that an inverse relationship between a given behavior and the resulting gene can now be established. A specialized software described in [12] allows the user to change the Y bits looking for the U vector to change so that it corresponds to a desired behavior. Moreover, conditions for the existence of a certain behavior may be easily translated into conditions imposed to the gene bits, as shown in an exemple in [4] for the case of semi-totalistic CA. In [9] three types of dynamic behaviors (i.e. imploding or I, exploding or E, and preserving or P) were associated to a condition imposed to the uncertainty vector **U.** They may be summarized as follows:

E: ***A profile is "exploding"*** (i.e. uncertainty spreads within the array or in other words local connectivity gives rise to global computation like random number generation or computation with gliders etc.) if there are at least two members (elements) of the profile with maximum value($U_k = U_l = 1$) and they are distant at more than n cells (i.e. $|k - l| \geq n$). Also, a profile is "exploding" if the above condition is not fulfilled but if the sum of uncertainties is larger than n

P: ***A preserving profile is a non-exploding one with at least n non-zero elements.*** Preserving behaviors imply that a computation takes place but dominated by the local connectivity (less complex, i.e. filtering).

I: ***An imploding profile is a non-exploding one but with less than n non-zero elements***. Such profiles are associated with a dynamics of the CA such that after a few (usually less than n) iterations, all cells are in the same sure state (all in 1 or all in 0 or oscillating). They correspond to period 1 and 2 behaviors.

In addition, one can analyze the U vector from the symmetry point of view. A gene may be declared as **S** (**symmetric**, when left side elements of **U** are equal to their right side elements – with U_5 in the middle) or **A** (**asymmetric**) and in this later case a degree of symmetry may be also computed [9]. It is noted that symmetric behaviors are less complex than asymmetric ones. For the 256 elementary CA a comparison between the classification in [10][11] and the one induced by the **U** profile gave a relatively consistent match following the rules given in the next table:

Class [10]:	1 and 2	3	4	5	6
U-based behaviors	I	PS	PA	ES	EA

In addition to Chua's classification, the UPT method reveals finer subdivisions within each behavioral domains and relationships to other interesting dynamic behaviors. For instance, most asymmetric (A) rules with "exploding" and "preserving" **U** profiles do have attractors that synchronize binary (the state of only one cell per iteration is sent to the receiver). Particularly interesting are those in the "exploding" case since they are usually chaotic. It was also found that binary synchronization is unlikely for symmetric (S) profiles. Also gliders are favored by a sum of uncertainties in the U profile that is close to *n.*

4. Selection of Desired Behaviors for the "1a5" CA

Let us consider the following problem: find a gene Y (or its associated ID) such that its corresponding "1a5" CA behavior is "exploding" and "asymmetric" (i.e. one that is the most complex).

A first solution is given next using a software that evaluates equations (2) (3) (4) interactively. We know that we are looking for a vector **U** which is asymmetric (**A**) and fulfills the condition **E**. One simple way to look for a gene is to enter an arbitrary decimal ID i.e. 22…2 until it is seen that the corresponding U profile fulfills the desired conditions. Such a situation is presented in Fig.2 from the panel of the software. The resulting profile is U= $[0\,8\,8\,14\,15\,14\,16\,8\,16]/32$ with a sum of uncertainties equal to 6.1875 (i.e. larger than 5.5). It is also clear that it is an asymmetric profile.

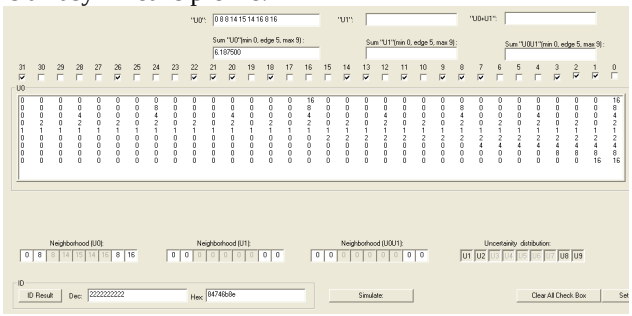

Fig.2. A particularly computed uncertainty profile for a given ID= 2222222222.

To test if the desired behavior is indeed in that class, one may employ the key "simulate" producing the evolution shown in Fig. 3 (time is on the vertical axis):

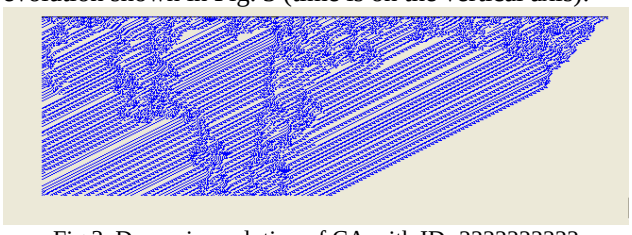

Fig.3. Dynamic evolution of CA with ID=2222222222 exhibiting a complex behavior (interacting gliders)

The same software allows to test for the binary synchronization property in a CA with the same ID and a given number of cells (time is now on the horizontal axis).

Three traces (upper is the Tx CA, middle the Rx CA and lower trace the difference of states between Tx and Rx) show that indeed such a phenomena occurs as predicted and expected for EA behaviors (Fig. 4).

Fig. 4: Dynamic evolution of a system formed of two master-slave CA with ID=2222222222. It is seen that after certain number of iterations the Rx CA is synchronized by one bit received from Tx CA.

Another solution to pick genes with desired behaviors consists in selecting a random pool of IDs (here 10,000) and use the UPT method to calculate the U profile for each of them. Note the very good speed of our method, where all profiles for 10,000 IDs were calculated in less than one second on a personal computer. A behavior space may be drawn as shown in Fig.5 after selecting two important parameters (cumulated uncertainty, i.e. the sum of all elements in U, and a symmetry index – detailed in [9]). Each point is assigned one of the behavioral classes as discussed before.

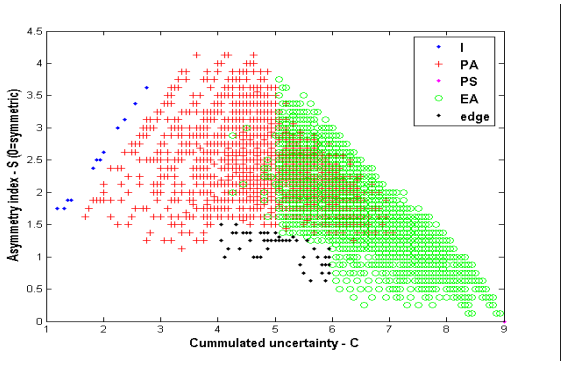

Fig.5. Distribution of 10000 IDs in a behavioral space determined by their U profile.

In addition to these classes it was observed that the most "interesting" behaviors hold for a low (but non zero) asymmetry index and in the edge between P and E behaviors. Since for each point in the behavioral space a list of IDs is given, looking for a desired behavior in such sub-regions (e.g. the black dots in Fig.5. located on the lower edge of the behavioral triangle domain) allow to rapidly locate IDs leading to desired CA dynamics. Simulations of such genes confirm the prediction. It is interesting to answer how many rules are assigned to each behavioral category. Using 10000 randomly selected IDs the conclusion is given in the next table, where the results for elementary CA (or "1a3" according to taxonomy in [4]) from [9] are given for comparison.

Behavioral class	I	PS	PA	ES	EA
1a5 CA	0.1%	0.3%	17.3%	1.6%	80.7%
1a3 CA	9.1%	20.4%	39.7%	15.9%	14.9%

Concluding, the UPT method may be conveniently applied to "1a5" type CA to identify desired behaviors. From the above table is clear that now most (more than 80%) of the CA cells are associated with complex dynamics instead of only 15% in the case of elementary ("1a3") CA.

Acknowledgement: This work was supported by the research grant ADBIOSONAR (No. 12079/2008, with CNMP, Romania).

References

[1] L.O. Chua, *CNN: A paradigm for complexity*, World Scientific, Singapore, 1998.

[2] L.O. Chua, "Passivity and complexity", *IEEE Trans. Circuits and Systems-I*, Vol. 46, No.1, pp. 71-82, 1999.

[3] R. Dogaru R, L.O. Chua LO, "Edge of chaos and local activity domain of FitzHugh-Nagumo Equation", *International Journal of Bifurcation and Chaos*, Vol. 8, No. 2, pp. 211-257, 1998.

[4] R. Dogaru, *Systematic design for emergence in cellular nonlinear networks – with applications in natural computing and signal processing*, Springer-Verlag, Berlin Heidelberg, 2008.

[5] R. Dogaru, S. Wang, H. Kim, "Fast Selection of Cellular Automata Genes based on Uncertainty Profiles of Cells", P*roceedings of the International Conference on Genetic and Evolutionary Methods* (GEM'08: July 14-17, Las Vegas, USA), pp. 216-221, 2008.

[6] R. Dogaru, "Hybrid Cellular Automata as Pseudo-Random Number Generators with Binary Synchronization Property", *Proceedings of the International Symposium on Signals Circuits and Systems* (ISSCS'09), Iasi Romania, July 2009, pp. 389-392.

[7] R. Dogaru, I. Dogaru, H. Kim, "Binary chaos synchronization in elementary cellular automata", *International Journal of Bifurcation and Chaos*, Vol. 19, Issue 9, pp. 2871-2884, 2009.

[8] R. Dogaru, I. Dogaru, H. Kim, " Chaotic Scan: A Low Complexity Video Transmission System for Efficiently Sending Relevant Image Features", in *IEEE Trans. on Circuits and Systems for Video Technology*, Vol.20, February 2010, pp. 317 – 321.

[9] R. Dogaru, "A Fast Method for Classification of Emergent Dynamics in Cellular Automata Based on Uncertainty Profiles", *in Journal of Control Engineering and Applied Informatics*, Vol 11, No 4 (2009), pp. 18-25.

[10] L.O. Chua, J. Guan,V.I. Sbitnev, J. Shin,"A Nonlinear Dynamics Perspective of Wolfram's New Kind of Science. Part VII: Isles of Eden," *Int. J. Bifurcation and Chaos*, 17(9), 2839-3012, 2007.

[11] L.O. Chua, K. Karacs, V.I. Sbitnev, J. Guan, S. Jinwook,"A Nonlinear Dynamics Perspective of Wolfram's New Kind of Science. Part VIII: More Isles of Eden," *Int. J. Bifurcation and Chaos*, 17(11), 3741-3894.

[12] R. Dogaru, I. Dogaru, H. Kim,"Representative Probability Tables for Fast Identification of Interesting Behaviors in Cellular Automata", in Proceedings of the International Symposium on Electrical and Electronics Engineering (ISEEE'08), Galati, Romania, pp. 7-12, 2008.