Non-Uniform Cellular Automata a Review

Daniel Camara

Department of Computer Science
University of Maryland
A.V. Williams Building
College Park, MD 20742

danielc@cs.umd.edu

Abstract **:** This tutorial intends to present an introduction to non-uniform cellular automata (CA). The difference between uniform cellular automata and non-uniform ones is that in uniform CA all cells share the same rule, or set of rules, in non-uniform CA this is not true, every cell can have a different sets of rules or even no rule at all. We will see here that this single difference increases the power of this kind of CA enabling them to perform computations impossible to their pairs in the same class of uniform CA. We will also see here some models and applications that show that non-uniform CA can perform complex tasks indeed being able even to perform universal computation.

1. Introduction

This tutorial presents the basic concepts behind non-uniform cellular automata and tries to show some interesting nuances about this kind of cellular automata. Cellular automata (CA) are dynamical systems in which the space and time are discrete. They were created by the mathematician John Louis von Neumann as a way to study the behavior of complex systems like models of self replication on biological systems [1].

A cellular automata, consists of a set of cells forming a regular grid, or a lattice. Each cell is characterized by a discrete set of integer variables that can have a finite number of values. The cell variables change simultaneously at discrete moments [2] following a set of rules. The canonical CA, as defined by von Neumann presents by definition:

- Massive parallelism
- Discrete behavior
- Decentralized control
- Local computations
- Simple cells
- The same set of rules on the entire grid

The three key characteristics, that make CA a powerful computational instrument are their massive parallelism, local computations and simple rules.

Fundamentally CA gives us a whole new, simplified and well known universe. This universe is far different of our universe that is complex and full of obscure relations. Indeed this new universe can be as simple or as complex as we want. We can, for example, input simple rules in parts of this universe and analyze the global impact of these rules on the entire universe. Cellular automata have many applications and have been used since as an alternative to differential equations [16] to a way to enhance landcover classification [17].

The main problem in using CA came from the difficulty in use their complex behavior to perform useful computations [3]. Depending on the problem it is very hard to make a set of cells to work together to solve a problem or to find a global solution.

From the parallelism point of view, cellular automata are different from regular parallel systems. CA do not split the problem into different processors and after that combine the solutions in order to calculate the final solution. Instead of it CA suggest a new approach in which complex behavior arises in a bottom-up manner from non-linear, spatially extended, local interactions [4].

On the next section of this tutorial we will introduce what are non-uniform cellular automata, after that we will talk about the dynamicity of the information on CA. Then we will present some scenarios in which non-uniform CA can be applied and finally, on the last section, we will present a conclusion to summarize the most important points observed during the tutorial.

2. What is a non-uniform cellular automata

Non-uniform cellular automata is a special case of cellular automata where not all cells present the same set of rules and these rules can change and evolve during the time. Actually, in some sense, we could say that uniform CA is a special case of non-uniform CA, once this is more general. Normally non-uniform CA still maintains the other characteristics of regular cellular automata. They are still massive parallel, discrete, simple and local with respect to the computations. The relaxation on the uniformity of the rules is the only difference between non-uniform CA and regular CA. However this difference, as we will see, increases the power of this kind of cellular automata if compared to regular ones.

Other difference, that is consequence of the non-uniformity, is the size of the rules search space. As each cell in non-uniform CA can have different rules the rules search space is extremely larger than in regular cellular automata. At first sight this seems to be an impediment, but in fact the increased search space size engenders new evolutionary paths, leading to high performance systems [3]. Moshe Sipper claims that the non-uniformity of this kind of cellular automata reduces the connectivity requirements among the cells enabling the decrease of the observed neighborhood [3]. This means that even observing a smaller neighborhood non-uniform CA can perform the same tasks uniform CA can do with a higher performance.

Non-uniform cellular automata are also capable of universal computation as proved by Sipper in [5]. Indeed on this particular paper it is proved that non-uniform CA are able to do universal computation with a 2-state, 5-neighbor cellular automata. With this same configuration Codd in [7], proof to be impossible to perform universal computations in uniform CA. Just this fact by itself is enough to perceive the enormous power of non-uniform cellular automata.

3. Dynamics of Information

Moshe Sipper presents in [5] an interesting discussion of under what conditions we can expect a complex dynamics of information to emerge spontaneously and dominate the behavior of a CA. His arguments are based on the works of Langton [6] and Codd [7], that in some way also approach this question.

Langton showed in [6] that the rule space in regular CA consists of two primary regimes of rules: periodic and chaotic. These two regimes are separated by a transition regime. Langton concluded in his work that nearly a critical phase transition the information processing can emerge spontaneously and came to dominate the dynamic of the entire system.

Codd in [7] proofed that uniform cellular automata, with one rule, two states and five neighbors cellular space is unable to perform universal computation. The central point of his argument is that either every configuration yields an unboundable propagation or every configuration yields a bounded propagation. In the context of Langton's work bounded propagation correspond to fixed point rules (class I) and unboundable propagation correspond either to periodic rules (class II) of chaotic ones (class III). Complex behavior (class IV) cannot be attained [5]. The classification is in accordance with Wolfram's classification [8]. The Figure 1 shows some examples of 1D CA to clarify how each class behaves. Being the class IV the one more likely to be the behavior of living things.

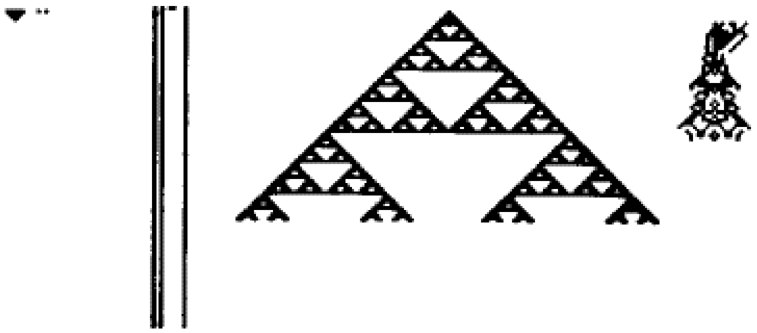


Figure 1- Classes of patterns generated by the evolution of cellular automata [8]. The leftmost is class I, the second class II, followed by a class III and finally the last one is a class IV CA

What Sipper claims at [5] is that small perturbations on the set of rules may be enough to cause changes on the observed world. These small changes could be enough to cause a transition from a class II to a class III or to class IV behavior. This change can happen independently of other regions of the virtual space.

The assumptions made by Sipper are based on the experiments performed by Miller [9], where he tries to emulate the primitive atmosphere of the earth which methane, ammonia, water and hydrogen. Miller arranged all this elements together in a controlled environment and submitted them to electrical sparks for a week. After this period simple amino acids were found in the system. In this way, in a simple and homogeneous environment, that does not support complex (class IV) behavior, in some point a spark causes a perturbation and a small number of "cells" change their rule. The main point here is that if this can occur on this controlled environment why not in cellular automata?

If this comparison is valid, and at least to us it makes a lot of sense, we can use non-uniform cellular automata to create artificial life in a level that goes far from the actual artificial life models and standards. With simple non-uniform CA we can evolve complex and complete virtual worlds. The possibilities are enormous, suppose for one instant that we could create these complex worlds. Probably, as the development of the rules are non deterministic, no two worlds would be exactly the same, so we could at some points verify what could happen if we "collide" some of these different worlds, or communities. As these worlds are digital, we could evolve them apart and join them in many different moments, verifying their evolving during the time. This kind of experiment could help us to understand the diversity of living elements in the nature or even how different communities, or modern organized societies, behave when exposed one each other.

4. Studied Models

There are a number of models and problems studied under the non-uniform CA point of view. These ones present here are just a small amount of all possible models to handle the rules in non-uniform CA. First we will observe how non-uniform, one-dimensional, cellular automata can be used to solve the density classification problem. After that we will study some models applied to handle the changes that occur on the rules space of non-uniform CA. Then we will discuss how to handle complex structures such as self-replication structures, and finally, as one lest example, we will present the worm experiment. On this experiment we will see that higher order behavior can appear based on simple local rules.

4.1.Density classification

Determining the density of the initial configuration of one-dimensional cellular automata it is a classic problem. The computational task is to decide if the initial CA configuration contains more than 50% of ones or not. If the initial CA configuration has more than 50% of ones the entire CA should evolve to have every state as 1 or to have all states as 0 otherwise.

Mitchell et al. point in [10] that this task is non trivial for a small radius of CA, being the small radius is in relation to the size of the grid. The main point of this observation is that the density is a global property based on the size of the grid. However with a small cell size we can just relay on the local information and iterations to find this global property.

The best algorithm to solve this problem was defined by Gacs Kurdyumov and Levin in [11]. Essentially their strategy (GKL) successively classify local densities with the locality range increasing over time. In regions of ambiguity a "signal" is propagated either as a checkerboard pattern in space-time or as a vertical white-to-black boundary [12]. Mitchell et al. in [12] approach this problem in a binary CA with 149 cells and a radius of 3, comparing their evolved solution to GKL. On Mitchell's work the best fitness values vary from 0.93 to 0.95, never being above 0.95. With the same fitness function GKL rule has a fitness ≈ 0.98 [4].

In [3] Sipper compares his work with the one of Mitchell et al. [12]. On his work he uses nearly the same parameters Mitchell uses and in addition Sipper also do an independent work with a smaller rule radius. In the same line of Mitchell's work, Sipper also creates a genetic algorithm to find a good set of rules. However his results are not as good as he expected. His best fitness is 0.92 when on in the Mitchell's work, with similar parameters and fitness evaluation the fitness vary from 0.93 to 0.95. Sipper attributes the low performance of the non-uniform CA to the large search space his genetic algorithm has to find good solutions. Since in this part of the work each cell contains one of 128 possible rules (r=3), and there are 149 cells, the search space size is $(2^{128})^{149} = 2^{19072}$.

A second part of Sipper's paper verifies the behavior of the non-uniform CA in the same problem, but now with a smaller radius. The work evaluates the problem with the smallest reasonable radius, r=1. This means that each cell knows only its own value and the values of its closest two neighbor. With this size radius the size of the search space decreases sensible, but it is still large, $(2^{8})^{149} = 2^{1192}$. However the size of the uniform CA search space is not, the search space to the uniform CA with r=1 is $(2^{8}) = 256$ rules. What Sipper did then was find the rules with the best fitness running every rule in 1000 different configurations. The results are showed at Figure 2. The highest fitness was 0.83 achieved by the rule 232. Thus this is the maximum performance one uniform r=2 cellular automata can reach. Now the question starts to be, can a non-uniform CA do better than this?

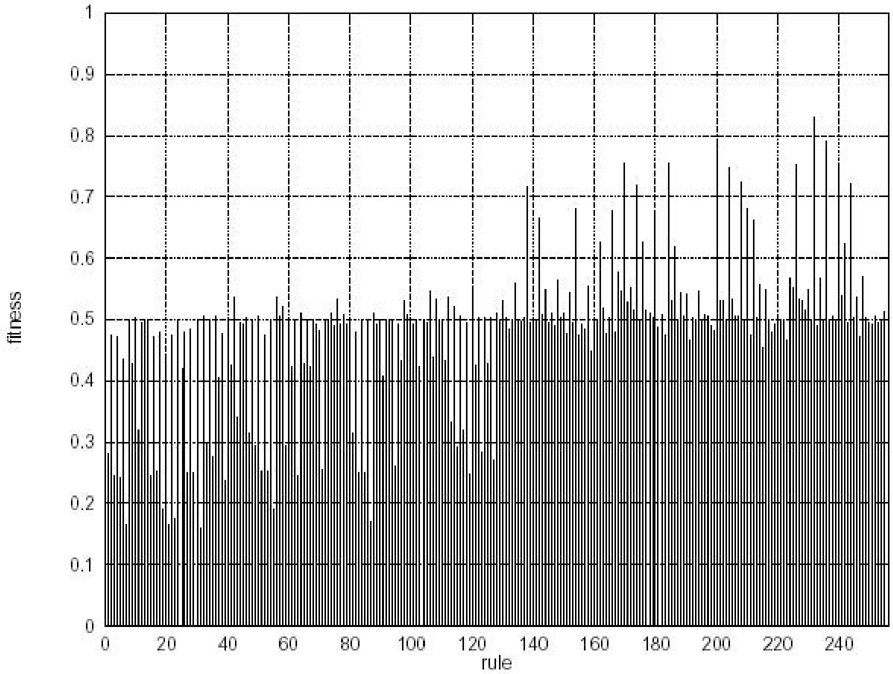


Figure 2 - Fitness results of all possible uniform, r = 1 CA rules

According to the results presented by Sipper the answer is yes. The best fitness with an r=1 CA was 0.93, lower than the 0.98 reached by the GKL algorithm, but much higher than the 0.83 achieved by the best non-uniform CA. Thus it is in the same level of those achieved by r = 2 and r = 3 CAs. This suggests that the non-uniformity reduces the connectivity requirements [3] and the use of smaller radiuses is made possible.

Sipper uses few rules, in deed he classify his work as a quasi-uniform CA. However the main point is that there are more than one rule at the cell space and one rule supports the other. Normally the evolved solutions present a dominant rule with other few cells with other dominated rules. The evolved, non-uniform CA puts these two kinds of rules together, in a way that one supports the other and they work together to reach the final objective.

4.2.Non-uniform automaton model

Now we will considerate two-dimensional cellular automata and different ways of evolve their rules space. In [14] Sipper presents some different CA models where the evolution of the cells do not occurs just in the cell value space, but it occurs also in the rules space also.

4.2.1. Evolution in rules space

The first model presents a non-uniform CA with nine neighborhood and binary states by cell. On this model a cell's rule may be regarded as a genotype whose phenotypic effect is achieved by rule application. A cell's genotype is reproduced if its phenotypic effect promotes fitness. What the author intends with this model is evolving it in the rule space starting from a random gene pool. The evolution of an unsuccessful cell's rule is accomplished by selecting one successful neighbor at random and copying its rule. If there is no successful neighbor then the unsuccessful rule remains unchanged.

The rules have now to have a criterion to determine their success or not. The criteria of success is defined as:

1. **Live -** A cell is considered to be successful if it attains a state of one (or live).
2. **Agree -** A cell is successful if it agrees, this means it is in the same state, with at least four of its neighbors.

An alternative, if we have non-binary success criteria, is to copy the most successful neighbor. Other factor to be observed is that the copy cannot be perfect, generating in this way mutations. In this specific model each cell contains just one rule. However, if each cell had a set of rules the imperfection at the copy could be done in terms of number of copied rules also. With this we could, in some way, simulate a crossover among the rules during the rules space evolution.

The problem with this model, that is called *Live and Agree,* is premature convergence. Live and agree admits many local minima, and all of them are equally valid, as far what concern to the evolutionary process. Sipper claims that this method can find the global minima if it uses a criterion called *parity.* The parity of a cell is equal to 0 if it has an even number of live neighbors and 1 otherwise. In such criteria a cell is successful if it is equal to the parity of its neighbors in the previous time step. However no real proof of experiment results are showed to support such claim.

Sipper also studied the behavior of this model when using a non-binary success criterion. The model he chose to be the criterion of success was the Iterated Prisoner's Dilemma (IPD) discussed by Axelrod in [13]. On this schema each cell plays IPD with its neighbors where ones represent cooperation and zeros represent defection. In this case a cell copies, with a small probability of mutation, the rule of the neighboring cell with the highest ranking, or total payoff. The total payoff is computed by adding the eight neighbors individual payoffs. This method, like life and agree, also admits many local minima and do not have a good performance. However what is interesting to notice about this experiment is that among the winning rules the average percentage of cooperation, ones, is 60%. This value is associated the quality of forgiveness, named a good quality for the game of life strategies. Indeed this value is near to the TIT-FOR-TAT strategy, named to be best strategy for solving the Iterated Prisoner's Dilemma, where the percentage of ones is 64%.

4.2.2. Formation of Complex Structures

The second model introduces a slightly enhancement over the previous model, however the author claims that this enhancement is simple enough to not break any CA rule.

In this model each cell is either vacant, containing no rule, or operational consisting of a finite state automaton, having the following characteristics:

1. Each cell can access its own state and the state of its immediate neighbors.
2. The cell can change its own state and the state of any immediate neighbor. If two cells try to change the same common neighbor at the same iteration, a criterion to resolve the concurrency must be defined. This criterion can be either random or deterministic.
3. The cell is able to copy its rule onto a neighboring vacant cell. A special case of copy is mobility, where the value, on the previous cell is erased. The same previous observation about concurrency is applied here.
4. A cell may contain a small number of different rules. However at a given moment only one rule is active and it determines the cell's behavior. This works in the same line of recessive genes, where the non-active, or recessive, genes are stored to be used in the future. Non-active rules may also be activated or copied onto a neighboring cell.

The main difference between this model, the previous one and regular CAs is that this model allows one cell to change the values of its neighbors. This is not a small change, but do not add much more complexity to the model and all actions are performed between neighbors. In this way the locality primitive is respected.

4.3.Self Reproduction

Using the last model Sipper presents, also in [14], some different ways to create self reproduction structures in non-uniform CAs. The first way he presents to create self reproduction is based on standard self replication machines where a loop structure projects an arm and this starts of the replication loop.

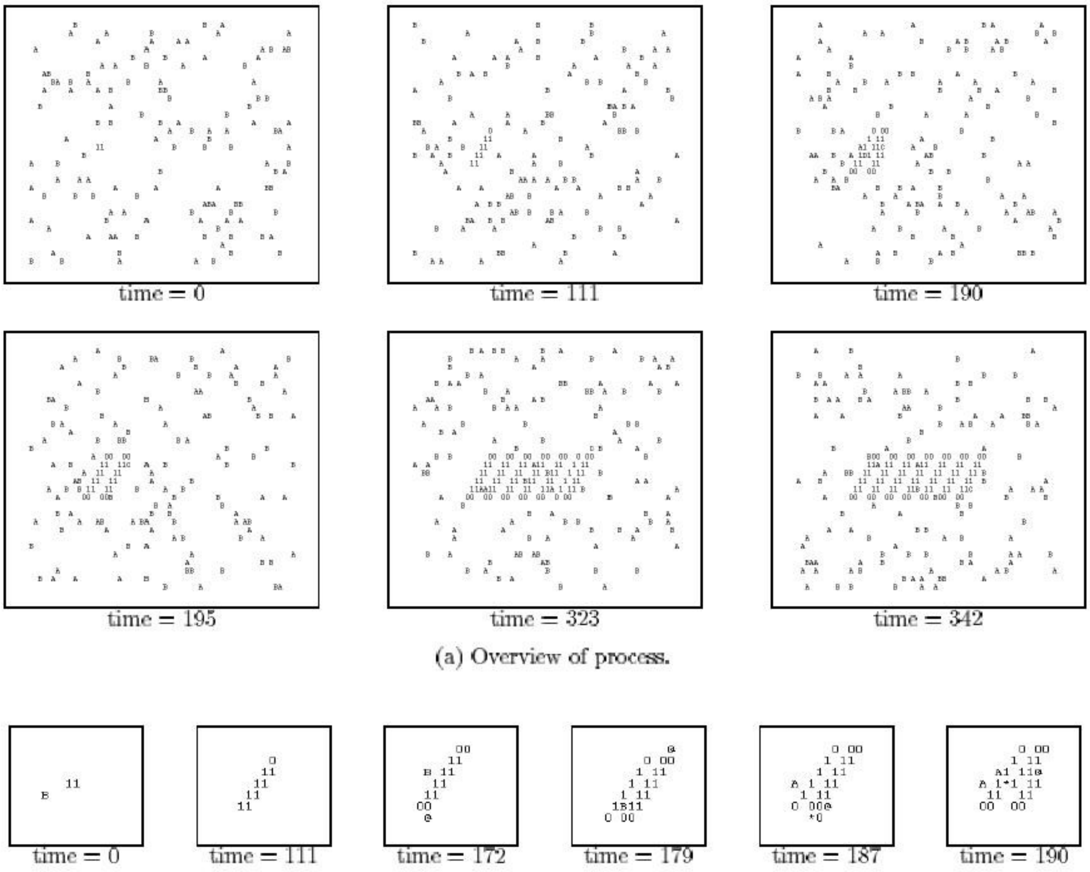
On this experiment the entire cell space starts clean, no cell contain any rule, with exception of just one single loop, a set of five cells that contains the same loop rule. The arm extends itself by copying its rule to neighbor cells. If a loop finds itself blocked by other loop it dies. Essentially this self replication loop works copying itself to other cells based on the status of its neighbor cells.

The second reproduction method, called Reproduction by Copier Cells, presents a set of active mobile cells acting over a set of static passive cells. This method has his roots in how the information flow in protein synthesis. On this process active tRNA structures translates passive mRNA structures into amino acid cells. Each tRNA cell matches one specific codon in the mRNA structure and synthesis one amino acid. This structure is represented by a fixed structure of ones and zeros and many “floating” X, Y and Z elements. These elements float through the space trying to find a right match, when this occurs, this floating structure creates the right complement to the static passive structure.

The last self reproduction structure discussed on the paper is the most complex one and it intends to form and replicate complex organisms. The cell space is randomly initialized with builder cells, represented by A, replicators cells represented by B and two core replicant cells represented by 1. These last cells will catalyze the building structure process.

Both cells A and B are randomly spread over the space and work together to create new and complex structures. The builder, A, when find a 1 structure it attaches another 1 on both sides of the growing structure. When a B cell are attached at one end of the growing structure the construction at that structure end stops.

Now follow Sipper explanation about the replication process observing the times showed in the Figure 3. When a B cell attaches itself to the upper end of a structure already possessing one zero a C cell is spawned, which travels down the length of the structure to the other end. If that end is as yet uncompleted the C cell simply waits for its completion (time 172). The C cell then moves up the structure, duplicating its right half which is also moved one cell to the right (time 179). Once the C cell reaches the upper end it travels down the structure, spawns a D cell at the bottom and begins traveling upward while duplicating and moving the right half (time 187). Meanwhile the D cell travels upwards between two halves of the structure and joins them together (time 190). This process is then repeated. The C cell travels up and down the right side of the structure, creating a duplicate half on its way up. As it reaches the bottom end a D cell is spawned which travels upward between two disjoint halves and joins them together. Since joining two halves occurs every second pass the D cell dies immediately every other pass (e.g. Time 195) [14].



(a) Overview of process.

(b) Zoom of intermediate stages. C cells are represented by @, D cells by $*$.

Figure 3 - Formation and reproduction of complex organisms[14]

Through this last model we can see clearly that non-uniform cellular automata, enable the creation of models in which different automatons work cooperatively to create new structures. This cooperative behavior among the cells can be very interesting when working in real and complex problems where we can broke the problem in small pieces that can be solved by different rules.

4.4.Mobility

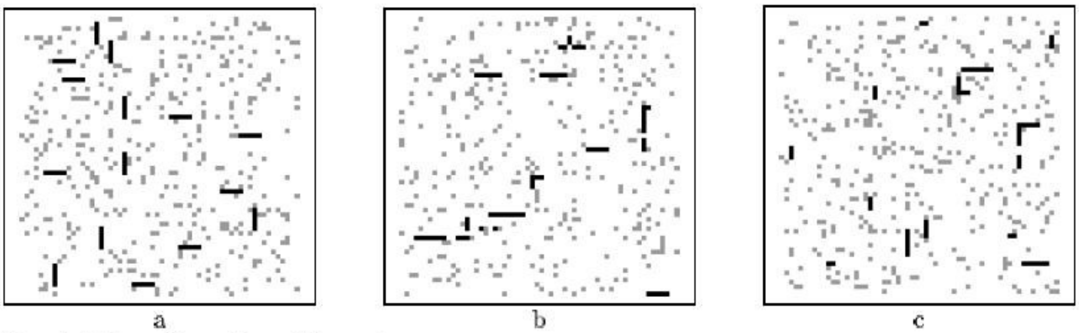
The last experiment we will present in this review is a worm-like structure that it is free to move through a grid of non-uniform Cellular Automata. Moshe Sipper presents this structure in [15] and the system consists with worms, which are active mobile structures represented by cells in state 1, and blocks, which are fixed structures in state 0. If a worm, that is a mobile rule, reaches a block it makes a 90° turn and continues its path. If we input more than one worm at the space at the same time, this simple scenario permits very interesting and complex high order behaviors.

When we have more than one worm at the grid the results when one worm meet other can be:

- One of them splits into two

- Both worms merge into one
- A worm looses part of its body
- Nothing happen to any one

In all cases the resulting worms behave exactly in the same way the normal worm behaves. It is interesting to observe the formation of such high order behavior because the rules act locally, not globally. Basically the programmed rules determine how the worm header and tail will behave, no rule was crated to handle collision between worms. The Figure 4 shows the worms experiment in some different moments.



(a) an initial configuration of the system.
(b), (c) system configurations after several time steps.

Figure 4 - A system consisting of several worms [15]

5. Conclusions

This tutorial intended to show a little bit about what are non-uniform Cellular Automata and what people have been doing with them. During this paper we sow that the only difference between uniform and non-uniform cellular automata is that the set of rules on the first one is the same to every cell and on the second no. Even though this is the only difference between the two CA forms, we sow that this increases the power of this new class of cellular automata. However they maintain unchanged the key CA features namely:

- Massive parallelism
- Discrete behavior
- Decentralized control
- Local computations

The main power of non-uniform CA is their cooperative behavior. Some times even if you have a good solution, but not a perfect one, you can use other rules to correct errors of your good solution. Interested readers can find deeper discussion about this subject in [3].

It is important to notice also that non-uniform and uniform CA, in the same category, have different capacities in solve problems. A good example of it is that while Langton proved in [6] that 2-state, 5-neighbor uniform CA are unable to perform universal computation, in [5] Moshe Sipper shows that non-uniform CA, with this same configuration, can perform universal computation.

However even non-uniform CA being clearly more powerful than uniform ones finding the right set of rules to apply on non uniform CA can be a draw back. The rules search space in most cases simply increases in a prohibitive way. Some times this search space is so huge that can be just impossible to find a good set of rules to solve a specific problem.

In short this new kind of CA has some very good and positive points, but has also some drawbacks that difficult its usage. We believe that non-uniform CA is a field in with there are still a lot of room for research, mainly in areas such as evolving knowledge and studding society behavior. We also believe that non-uniform cellular automata provide an effective way to explore and study artificial life in its more variable forms.

6. References

[1] J.von Neumann, Theory of Self-Reproducing Automata, University of Illinois Press, Illinois, 1966. Edited and completed by A. W. Burks

[2] V K Vanag, Study of spatially extended dynamical systems using probabilistic cellular automata, Physics – Uspekhi 42 (5) 413-434, 1999

[3] M. Sipper, Co-evolving non-uniform cellular automata to perform computations, Physica D, vol. 92, pp. 193-208, 1996

[4] M. Mitchell, James P. Crutchfield, and Peter T. Hraber, Evolving cellular automata to perform computations: Mechanisms and impediments, Physica D, 75:361-391, 1994

[5] M. Sipper, Quasi-uniform computation-universal cellular automata,ECAL'95: Third European Conference on Artificial Life, F. Morán, A. Moreno, J. J. Merelo, and P. Chacón, Eds., Heidelberg, 1995, vol. 929 of Lecture Notes in Computer Science, pp. 544-554, Springer-Verlag

[6] Christopher Langton, Langton, et al editors, Artificial Life II, 41-91, Redwood City, CA, 1992, Addison-Wesley

[7] E. F. Codd, Cellular Automata, Academic Press, New York, 1968

[8] Stephen Wolfram, Cellular Automata as Models of Complexity, Nature, 311, 419-424, October, 1984

[9] S. L. Miller, A production of amino acids under possible primitive earth conditions. Science, 117:528-529, May 1959

[10] Melanie Mitchell, J. P. Crutchfield, and P. T. Hraber, Dynamics computation and the "edge of chaos": A re-examination. In G. Cowan, D. Pines and D. Melzner, editors, Complexity: Metaphors, Models and Reality, pages 491-513, Addison-Wesley, Reading, MA, 1994

[11] P. Gacs, G. L. Kurdyumov and L. A. Levin. One-dimensional uniform arrays that wash out finite islands. Problemy Peredachi Informatsii, 14:2-98,

1978

[12] M. Mitchell, P. T. Hraber, and J. P. Crutchfield. Revisiting the edge of chaos: Evolving cellular automata to perform computations. Complex Systems, 7:89- 130, 1993

[13] R. Axelrod, The Evolution of Cooperation, New-York: Basic Books, Inc. 1994

[14] Moshe Sipper, Non-Uniform Cellular Automata: Evolution in Rule Space and Formation of Complex Structures, Artificial Life IV, R. A. Brooks and P. Maes (eds.), pages 394-399, 1994. copyright The MIT Press 1994.

[15]Moshe Sipper, Studying Artificial Life Using a Simple, General Cellular Mode, Artificial Life Journal, Volume 2, Number 1, pages 1-35, 1995. Copyright, The MIT Press 1995.

[16] Toffoli T., Cellular automata as an alternative to Differential equations, in Modelling Physics, Physica 10D, 1984

[17] Joseph P. Messina, Stephen J. Walsh, Greg Taff, and Gabriela Valdivia, The Application of Cellular Automata Modeling for Enhanced Land Cover Classification in the Ecuadorian Amazon, IV International Conference on GeoComputation, Fredericksburg, VA, USA, 25-28 July 1999