A Genetic Algorithm to Study a P3 Non-trivial Collective Task

F. Jiménez-Morales[1] and J. L. Guisado[2]
[1]*Dep. Física Materia Condensada, University of Sevilla, Avda. Reina Mercedes s/n, Sevilla, Spain*
[2]*Departamento de Arquitectura y Tecnología de Computadores, Universidad de Sevilla, Sevilla, Spain*
{*jimenez, jlguisado*}*@us.es*

Keywords: Cellular Automata, Genetic Algorithm, Non-trivial Collective Behaviour, Period-3.

Abstract: Here we report new results of a genetic algorithm (GA) used to evolve one dimensional Cellular Automata (CA) to perform a P3 non-trivial collective behavior task. For this task the goal is to find a CA rule that reaches one final configuration in which the concentration of active cells oscillates among three different values. Though the majority of the best evolved rules belong to the II Wolfram's class, the GA also finds rules of the III and IV classes. The different computational mechanisms used by each rule to synchronize the entire lattice are analyzed by means of the spatio-temporal patterns generated.

1 INTRODUCTION

Many natural and man-made systems that consist of a decentralized collection of simple units with local interaction can display complex emergent behavior. The term emergent behavior refers to the appearance in the system's temporal behavior of information-processing capabilities that are not explicitly represented in the system's elementary components. These decentralized systems have many advantages when compared with central control systems and there is both a practical and a theoretical interest in the study of how to design such sytems and their interactions in order to produce a useful emergent behavior.

One of the most popular methods for the analysis of spontaneous emergence of ordered behavior in spatially extended systems that are locally coupled is the use of cellular automata (CA) and genetic algorithms (GA) developed by the "Evolutionary Cellular Automata (EVCA)" group of M. Mitchell, J.P. Crutchfield and their colleagues (Mitchell et al., 1993; Mitchell et al., 1994; Crutchfield and Mitchell, 1995; Das et al., 1994; Das et al., 1995). In their studies a CA performing computations means that the input to the computation is seen as the initial state of each cell and the output is the final state reached by the iterations. They analyzed two computational tasks such as the density classification and the synchronization tasks in one dimensional binary CA. For the density classification task the goal is to find a CA rule that determines whether or not the initial configuration contains more cells in state 1 than cells in state 0. If it does, the whole lattice should eventually iterate to the fixed point configuration of all cells in state 1; otherwise it should eventually iterate to the fixed-point configuration of all 0s. For the synchronization task a successful CA will reach a final configuration in which all cells oscillate between all 0s and all 1s on successive time steps.

In this paper we focus in another computational task for one dimensional CA such as the period-3 task in which the goal is to find a rule that, starting from a random initial condition, reaches one final configuration in which the entire lattice oscillates among three different states. Originally (Jiménez-Morales, 1999; Jiménez-Morales, 2000) this task was successfully used to study the appearance of non-trivial collective behavior in three dimensional CA and in (Jiménez-Morales et al., 2002) we made a preliminary study in one dimension, also the computational mechanics of an evolved rule was fully described in (Jiménez-Morales and Tomassini, 2004). In this paper we use the same evolutionary process to evolve a population of one dimensional CA to perform the P3 task but we report here other answer rules that show irregular and chaotic patterns.

The paper is organized as follows: section 2 describes the P3 task, section 3 outlines the characteristics of our genetic algorithm, the results are shown in section 4 and finally the conclusions are shown in section 5.

Table 1: Four of the best evolved rules, the rule table hexadecimal code, the type of non-trivial collective behavior, the Langton's parameter, the fitness function versus the highest one and the Wolfram's class. To recover the 128-bit string giving the output bits of the rule table, expand each hexadecimal digit to binary. The output bits are then given in lexicographic order. The arrangement of neighbours in the genetic algorithm is $|s_{i-3}|s_{i-2}|s_{i-1}|s_i|s_{i+1}|s_{i+2}|s_{i+3}|$.

Symbol	Rule Table Hexadecimal Code	NTCB	λ	$F(\phi)/F(\phi_d)$	Wolfram's Class
ϕ_a	90001c50-27307066-0820a852-25650800	P3	0.3203	0.12	II
ϕ_b	21088418-01091108-41038844-10c18080	P3(QP3)	0.2109	0.41	II
ϕ_c	f193800-c06b0eb0-e000461c-80659c11	P3	0.3359	0.05	IV
ϕ_d	146157d1-fbb53fec-7dfbeffc-eaf0fa28	P3	0.6250	1.0	III

2 THE COMPUTATIONAL P3 TASK

2.1 Cellular Automata

Cellular automata are discrete dynamical systems consisting of an array of cells, each of which can be in one of a finite number k of possible states, updated synchronously in discrete time steps, according to a local, identical interaction rule. Here we will only consider Boolean automata for which the cellular state $s \in \{0,1\}$. In this work the regular cellular grid is one-dimensional, i.e. $d = 1$. The state of a cell at the next time step is determined by the current states of the cell itself and of a surrounding neighborhood of cells and is given by:

$$s_i^{t+1} = \phi(s_{i-r}^t ..., s_i^t, ... s_{i+r}^t) \qquad \phi : k^{2r+1} \longrightarrow k \qquad (1)$$

where s_i^t denotes the value of site i at time t, ϕ represents the local rule of evolution, and r is the CA radius i.e., the number of neighbors (cells) on either side of a given cell. The configuration of the entire lattice at a given time t can be described by $\mathbf{s}^t = (s_0^t, s_1^t, . . . , s_{N-1}^t)$, where N is the lattice size. Periodic boundary conditions $s_{N+i}^t = s_i^t$ are used. To study the spatio-temporal patterns generated by the CA dynamics, a global update rule Φ can be defined which applies in parallel to all the cells $\mathbf{s}^{t+1} = \Phi(\mathbf{s}^t)$; and an ensemble operator $\mathbf{\Phi}$ which operates on sets of lattice configurations $\{\mathbf{s}^t\}$ (Wolfram, 1994).

2.2 Non-trivial Collective Behavior

An interesting situation of emergent behavior in CA is found with the appearance of non-trivial collective behavior (NTCB). As CA are governed by local interactions and subjected to noise, it was expected that any global observable, such as the concentration of activated cells $c(t) = \frac{1}{N}\sum_i^N s_i(t)$ would show a trivial time dependence in the limit of infinite size (Bennet et al., 1990). But several exceptions to this have been found. The most remarkable one is a quasiperiod three behavior (QP3) that exhibits the concentration of rule-33 automaton in d=3 and other CA in higher space dimensions (Chaté and P.Manneville, 1992). This behavior is neither transient nor due to the finite size of the lattice and has been obtained for deterministic and probabilistic rules. Several attempts have been made to understand its phenomenology and have addressed the possible mechanisms by which this puzzling collective behavior emerges but yet there is not any answer to the question of how NTCB can be predicted from the local rule. Then in (Jiménez-Morales, 1999) we proposed using a genetic algorithm to the search in d=3 of CA rules with QP3 periodicity and in (Jiménez-Morales and Tomassini, 2004) the same algorithm was used in d=1. Quasiperiodicity is the most interesting non-trivial collective behavior but in d=1 can only be observed under some specific conditions. Here we will focus mainly in period-3 collective behavior. As stated before for this P3 task the goal is to find a rule for wich the concentration is oscillating among three different values, i.e., $c(t) = c(t+3)$.

3 THE GENETIC ALGORITHM

Our GA begins with a population of $P = 20$ randomly generated chromosomes, listing the rule-table output bits in lexicographic order of neighborhood patterns. We consider binary CA with periodic boundary conditions. Each CA is represented by a bit string delineating its rule table ϕ, containing the output bits for all possible neighborhood configurations. The bit string is of size $2^7 = 128$, resulting in a huge space of 2^{128} possible rules. The fitness evaluation for each CA rule is carried out on a lattice of N cells starting from a random initial condition of concentration 0.5. After a transient time of $N/2$ time steps, we allow each rule to run for a maximum number of M iterations. The values of concentration are assembled in groups of 4 consecutive values, noted as c_1 to c_4, and the fitness function $F(\phi)$ is defined by:

$$F(\phi) = \frac{4}{M} \sum_{i}^{M/4} \frac{1}{2} abs[(c_2 - c_1)(c_4 - c_2) - (c_3 - c_2)(c_3 - c_1)]_i \tag{2}$$

The rule's fitness $F(\phi)$ is taken from a geometrical point of view and it is an average area in the iterative map, i.e. the graph of $c(t+1)$ versus $c(t)$. In this iterative map the area of a period-2 behavior is too small, almost 0, the area of a noisy period-1 and the area of an intermittent P2 is higher than that of a P2 and finally a P3 and quasiperiod-3 behaviors have the highest values.

In each generation: (i) $F(\phi)$ is calculated for each rule ϕ in the population. (ii) The population is ranked in order of fitness. (iii) A number $E = 5$ of the highest fitness ("elite") rules are copied without modification to the next generation. (iv) The remaining $P - E = 15$ rules for the next generation are formed by single-point crossover between randomly chosen pairs of elite rules. The off-springs from each crossover are each mutated with a probability $m = 0.05$. This defines one generation of the GA; it is repeated $G = 10^3$ times for one run of the GA.

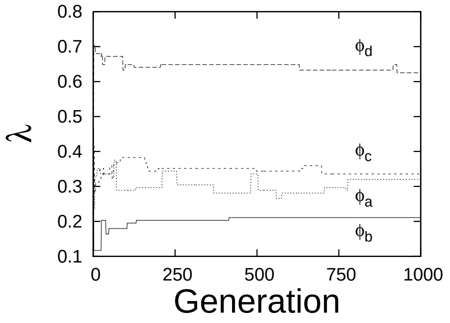
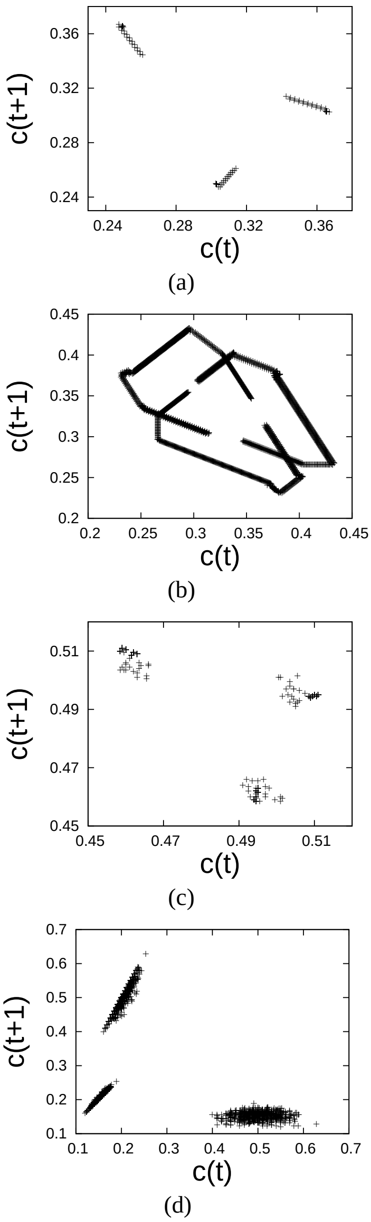


Figure 1: The λ parameter of the best rule in each generation versus the generation for the runs in which ϕ_a, ϕ_b, ϕ_c and ϕ_d were encountered.

4 RESULTS

We performed more than 200 different runs of the GA each with a different random-number seed. In $d = 1$ the GA is able to find many rules with the desired behavior, about 10% of the runs ended up with a rule that showed a P3 collective behavior or a quasiperiod-3. Table 1 shows four of the best evolved rules, the rule table hexadecimal code, the λ parameter (defined as the fraction of nonzero output bits in the rule table), the non-trivial collective behavior observed, and the corresponding Wolfram's class (Wolfram, 1984).



(a)

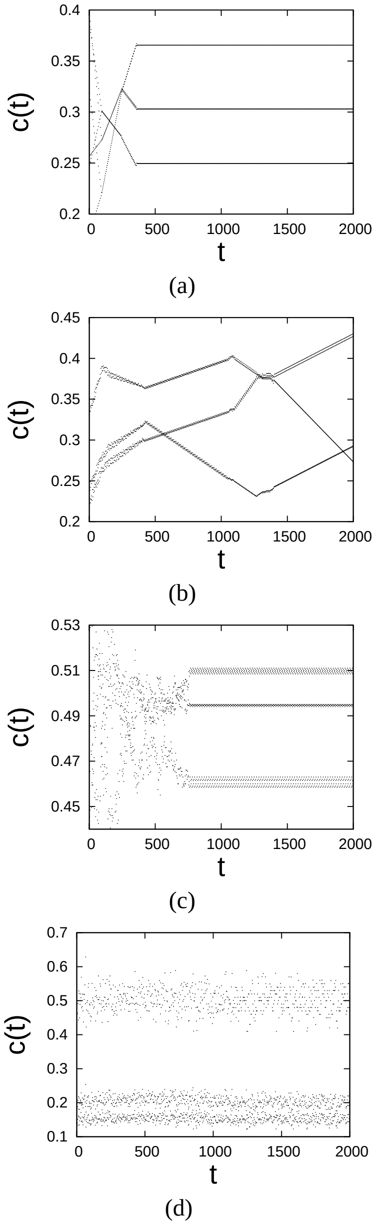
(b)

(c)

(d)

Figure 2: The iterative map for the rules: (a) ϕ_a, (b) ϕ_b, (c) ϕ_c and (d) ϕ_d. Lattice size is $N = 2000$ cells. Transient times are discarded. The collective behavior for the rules ϕ_a, ϕ_c and ϕ_d is a period-3 behavior. Rule ϕ_c shows two triangular objects that corresponds to a quasiperiod-3 collective behavior.

Figure 1 shows the λ parameter of the best evolved rule in each generation versus the generation. The initial population of candidate rules have a random λ between $(0,1)$. For all runs of the GA in the first 250 generations the best selected rule has a λ which has small differences with the best evolved rule. It has



(a)

(b)

(c)

(d)

Figure 3: The time series of the concentration for the rules (a):ϕ_a, (b):ϕ_b, (c): ϕ_c and (d):ϕ_d. Lattice size is $N = 2000$. The initial condition is a random configuration with concentration 0.5.

been suggested (Langton, 1990) that there is a relationship between the ability of a CA to show complex behavior and the λ parameter. The basic hypothesis was that λ correlates with computational capability in that rules capable of complex computation must be or are most likely to be found near some critical value ($\lambda_c \approx 0.26$) and also that these rules belong to the Wolfram's class IV. Our GA has selected rules with λ that goes from 0.21 to 0.62 and the majority of them belong to the II class and some others to the III class. Only in one run the GA encountered a rule of the IV class, ϕ_c, but it has the lowest fitness. The results from our evolutionary process do not show any kind of critical value of λ (an "edge of chaos") for the best fitness rules. In this sense the P3 task confirms the same result as it was obtained by (Mitchell et al., 1993) for the density classification task.

The behavior of the best evolved rules is shown in three plots: a) Figure 2 that shows the iterative map, i.e. the concentration at time $t+1$ versus the concentration at time t. In this iterative map transients points are discarded to better define the collective behavior. b) Figure 3 that shows the time series of the concentration from the starting time $t = 0$. And finally c) Figure 4 that shows a space-time diagram of 258x300 points after a transient of 200 time steps.

Rules ϕ_a and ϕ_c show a P3 behavior that can be seen as three well defined clouds of points in the iterative map and in the time series of the concentration, after a transient time, as three different constant values. The time series of the concentration for rule ϕ_b is different and shows three branches that interact and mix among them. This behavior lasts for very long times and as the lattice size increases the behavior is better defined. Now the iterative map rather than three clouds of points is a triangular object and the collective behavior of rule ϕ_b corresponds to a cuasiperiod-three (QP3) behavior (Jiménez-Morales and Tomassini, 2004). The rule with the highest value of the fitness function is ϕ_d. Its concentration also oscillates among three values but in a noisy fashion and its attractor in the iterative map consists in a fuzzy 3-cycle behavior.

Under the fitness function $F(\phi)$ in $d = 1$ the evolutionary process has selected rules that starting from a random initial condition synchronize the whole system to a state in which the concentration oscillates in a three-state cycle. But the way in which the four rules obtain the synchronization is quite different from rule to rule. To understand those mechanisms we can use the tools of the "computational mechanics" developed by Crutchfield and Hanson (Hanson and Crutchfield, 1997). This point of view describes the computation embedded in the CA space-time configuration in terms of domains, defects and defect interactions.

Figure 4a,b shows a space-time diagram of the rules ϕ_a and ϕ_b. Time starts after a transient time of 200 time steps and goes from up down and space is displayed on the horizontal axis. In the patterns generated by both rules there are some easily recognized spatio-temporally periodic background -the domains- on which some dislocations move. A domain, $\mathbf{\Lambda}$, is a set of configurations with two properties: (i) temporal

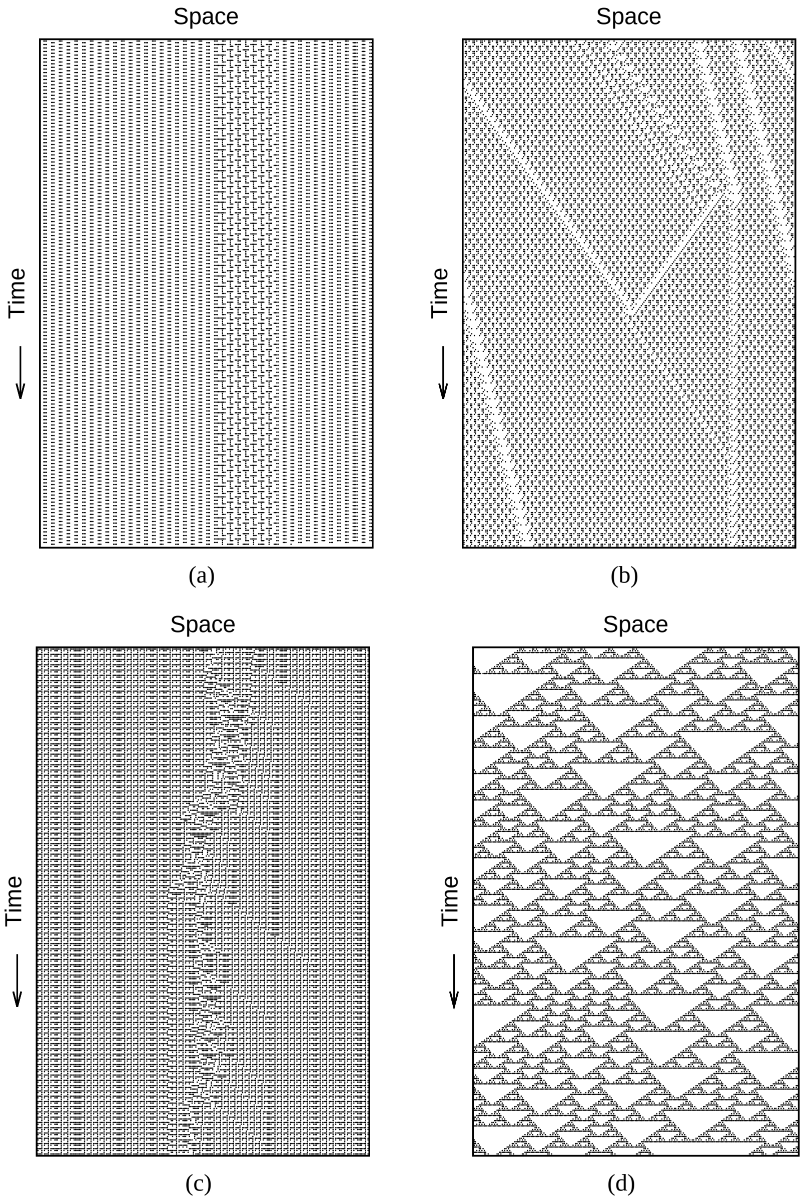


Figure 4: Space-Time diagram of four answer CA rules for the computational P3 collective task. Lattice size of 258 cells. Starting from a random initial condition with $c(0) = 0.5$ the first 200 time steps are discarded. It is shown a window of 258x300 points. (a) ϕ_a. This rule of the II class shows a regular and periodic final state with two fixed domains; (b) ϕ_b. This is also a rule of the II class with a periodic background in which defect cells or particles propagate from distant parts of the lattice; (c) ϕ_c. It can be seen long-lived and irregular propagating defect cells characteristic of the IV class. (d) ϕ_d. This is a chaotic rule (III class) that has the highest fitness.

invariance, which means that for some finite temporal period p, $\mathbf{\Phi}^p \mathbf{\Lambda} = \mathbf{\Lambda}$; and (ii) spatial homogeneity. In the simplest case a domain consists of a set of cells in the space-time diagram that are always repeated; for example, the domain for rule ϕ_b is shown in Table 2. If over a long time all the cells of the space-time diagram are in the domain then the concentration of activated cells will be oscillating among three values $1/2$, $1/3$ and $1/6$. Displacements of the domains along the temporal or the spatial axis give place to other domains and at the boundaries between them appear defect cells or particles that propagate among the entire lattice. These defect cells are spatially localized structures and time-invariant for rules ϕ_a and ϕ_b.

But the space time diagram of rules ϕ_c and ϕ_d (Figure 4c-d) is not so simple as the previous ones.

Rule ϕ_c also has a periodic background that fulfils the requirements of having a period-3 oscillating concentration, but the main difference with ϕ_a and ϕ_b is the way the global coordination is acquired. The defect cells for rule ϕ_c are irregular propagating structures, which is a characteristic of rules that belong to the IV Wolfram's class. And finally are the patterns generated by the rule ϕ_d where there is a lack of regularity or periodic background. While rules ϕ_a, ϕ_b and ϕ_c evolve towards a final state in which the concentration reaches a well determined three state, the three values of the concentration for rule ϕ_d show a noisy behavior (Figure 3d). Rule ϕ_d is a chaotic rule and its behavior may be interesting from an statistical point of view, though this is out of the scope of this work.

Table 2: Domain for rule ϕ_b and the concentration.

	Space						
	1	1	0	1	0	0	$c_1 = 1/2$
T	1	1	0	0	0	0	$c_2 = 1/3$
i	0	1	0	0	0	0	$c_3 = 1/6$
m	1	0	0	1	1	0	$c_1 = 1/2$
e	0	0	0	1	1	0	$c_2 = 1/3$
	0	0	0	0	1	0	$c_3 = 1/6$

5 CONCLUSIONS

The study of non trivial collective behavior in CA has suggested a new computational task like the P3 task. A genetic algorithm, with the appropriate fitness function, has been able to find answer rules that have regular repeating space-time configurations. These rules, belonging to the Wolfram's class II, attained the global coordination among the entire lattice cells with propagating structures –particles– that interact among them until a regular and final state is reached. In the patterns generated by these rules it can be distinguished domains and particles like in the density and the synchronization tasks. And the global coordination can be explained using the tools of the "computational mechanics" originally developed by the EVCA group of Crutchfield and Mitchell.

But our GA has also found some rules (class III and IV) for which the space-time diagrams do not show a regular pattern. The emergence of a global computation by these chaotic and complex rules is very interesting and points out the existence of other synchronization strategies not well understood yet. We think that many of the results obtained in tasks like this one can provide a useful information to establish a more general framework to study the emergence of global coordination in one dimensional CA.

REFERENCES

Bennet, C. H., Grinstein, G., He, Y., Jayaprakash, C., and Mukamel, D. (1990). Stability of temporally periodic states of classical many-body systems. *Phys. Rev. A*, 41:1932–1935.

Chaté, H. and P.Manneville (1992). Collective behaviors in spatially extended systems with local interactions and synchronous updating. *Progress Theor. Phys.*, 87(1):1–60.

Crutchfield, J. P. and Mitchell, M. (1995). The evolution of emergent computation. *Proceedings of the National Academy of Science U.S.A.*, 92:10742–10746.

Das, R., Crutchfield, J. P., Mitchell, M., and Hanson, J. E. (1995). Evolving globally synchronized cellular automata. In Eshelman, L. J., editor, *Proceedings of the Sixth International Conference on Genetic Algorithms*, pages 336–343, San Francisco, CA. Morgan Kaufmann.

Das, R., Mitchell, M., and Crutchfield, J. P. (1994). A genetic algorithm discovers particle-based computation in cellular automata. In Davidor, Y., Schwefel, H.-P., and Männer, R., editors, *Parallel Problem Solving from Nature—PPSN III*, volume 866, pages 344–353, Berlin. Springer-Verlag (Lecture Notes in Computer Science).

Hanson, J. E. and Crutchfield, J. P. (1997). Computational mechanics of cellular automata: An example. *Physica D*, 103:169–189.

Jiménez-Morales, F. (1999). Evolving three-dimensional cellular automata to perform a quasiperiod-3(p3) collective behavior task. *Phys. Rev. E*, 60(4):4934–4940.

Jiménez-Morales, F. (2000). The evolution of 3-d c.a. to perform a collective behavior task. In Miller, J., Thompson, A., P.Thompson, and Fogarty, T. C., editors, *Evolvable Systems: From Biology to Hardware ICES-2000, LNCS 1801*, pages 90–102, Berlin. Springer-Verlag.

Jiménez-Morales, F., Mitchell, M., and Crutchfield, J. (2002). Evolving one dimensional cellular automata to perform non-trivial collective behavior task: One case study. In Sloot, P., Tan, C. K., Dongarra, J., and Hoekstra, A., editors, *Computational Science ICSS-2002, LNCS 2331*, pages 793–802, Berlin. Springer-Verlag.

Jiménez-Morales, F. and Tomassini, M. (2004). Globally synchronized oscillations in an one-dimensional cellular automaton. *Int. Jour. Modern Physics C*, 15(3):409–425.

Langton, C. G. (1990). Computation at the edge of chaos: Phase transitions and emergent computation. *Physica D*, 42:12–37.

Mitchell, M., Crutchfield, J. P., and Hraber, P. T. (1994). Evolving cellular automata to perform computations: Mechanisms and impediments. *Physica D*, 75:361 – 391.

Mitchell, M., Hraber, P. T., and Crutchfield, J. P. (1993). Revisiting the edge of chaos: Evolving cellular automata to perform computations. *Complex Systems*, 7:89 – 130.

Wolfram, S. (1984). Universality and complexity in cellular automata. *Physica D*, 10:1.

Wolfram, S. (1994). *Cellular Automata and Complexity*. Addison-Wesley, USA, first edition.