Applied Mathematics & Information Sciences
*An International Journal*

http://dx.doi.org/10.12785/amis/090417

# Graph Dynamical Systems with General Boolean States

*Juan A. Aledo, S. Martinez and Jose C. Valverde*[*]

Department of Mathematics, University of Castilla-La Mancha, 02071 Albacete, Spain

Received: 18 Oct. 2014, Revised: 18 Jan. 2015, Accepted: 19 Jan. 2015
Published online: 1 Jul. 2015

**Abstract:** In this work we introduce a wide generalization of dynamical systems over graphs, by considering that the states of the entities can take values in an arbitrary Boolean algebra with $2^p$ elements, $p \in \mathbb{N}$, $p \geq 1$. Then the orbit structure of these more general parallel dynamical systems over undirected graphs where the evolution operator is an arbitrary maxterm or minterm is analyzed. Finally, we also study the cases of parallel dynamical systems whose evolution update is defined by means of independent local Boolean functions.

**Keywords:** Graph Dynamical Systems, Orbit Structure, Boolean Algebras, Boolean Functions.

## 1 Introduction

A *graph dynamical system* (GDS) is a dynamical system constructed over a graph whose vertices, named *entities*, can have different states, such that all these states together at a given time constitute a state of the system which can evolve thanks to an updating scheme. The states of the vertices are commonly modeled by the Boolean values 0 and 1, while the updating scheme consists of as many local functions as vertices and a series of rules that indicate the order in which the local functions act.

When all the local functions act synchronously the system is called *parallel* (PDS) [1,2,3,4,5,10]. In contrast, when the local functions follow an order to act, the system is called *sequential* (SDS) [10,20].

In the specific literature, other related topics appeared previously, as *cellular automata* (CA) [15,22,25,26] and *Boolean networks* (BN) [16,17], which are, in fact, particular cases of GDS.[1].

CA, when finite, can be considered as a special kind of PDS by considering cells as entities. Nevertheless, CA are restricted cases of PDS in several ways. First of all, for a CA seen as a PDS, the dependency graph, which is derived from the lattice and the neighborhood structure, is regular, whereas the graph of a general PDS is arbitrary. Secondly, CA have a fixed local function or rule, associated to every cell, while general PDS can have distinct local functions to update different entities, which can be the restriction of a global one (see [1,10]) or independently defined (see [3]). Thus, general PDS can have more involved update schemes.

CA are also updated in a parallel or synchronous manner by applying local functions on a subset that contains the (state value of the) cell. Nevertheless, in the last few years some extensions of the concept of CA, considering sequential or asynchronous updating, have appeared in the literature (see [11,19,23]). In fact, the concept of SDS constitutes a generalization of such a CA extension.

BN are a generalization of (finite) Boolean CA but, at the same time, a particular case of GDS by considering nodes as entities. One of the main differences with CA is that, in BN, the state of each node is not affected necessarily by its neighbors, but potentially by any node in the network. Thus, the uniform structure of neighborhood in CA disappears. However, some homogeneity remains, since each node is affected by $k$ connections with other (or the same) entities. This homogeneity makes BN a particular case of GDS, since in GDS connections can be totally arbitrary. Another important difference between BN and CA is that, for BN, local Boolean functions of $k-$variables are generated randomly, which provides a different update schedule for each entity. This idea has been carried out and extended for PDS in two directions. Firstly, as can be seen in [1], local Boolean functions acting on each entity can have different number of variables (what cannot occur for BN);

[1] The abbreviations GDS, PDS, SDS, CA and BN will be used for the singular and plural forms of the corresponding terms, since it seems better from an aesthetic point of view.

[*] Corresponding author e-mail: jose.valverde@uclm.es

© 2015 NSP
Natural Sciences Publishing Cor.

and secondly, they can be totally independent for each entity [3].

GDS, as a concept that generalizes the aforementioned ones, is relatively young and unexplored. In fact, the first ideas appeared in [6], which constituted an important step in the development of the mathematical foundations for the theory of Computation. In this work, sequentially updated cellular automata (SCA) over arbitrary graphs are employed as a paradigmatic framework. This first work was followed by [7], [8] and [9], where the authors developed this theory, analyzing the asymptotic behavior of such mathematical models. Later, many other works have appeared in order to describe the behavior of these dynamical systems (see [1, 2, 3, 4, 5, 10]) and also as applications of them to other questions (see [12, 14, 13, 18]).

In all of these works, the entities in the model can only have two state values, i.e., each entity can be either activated or deactivated. This is usually modeled by means of Boolean variables $x_i \in \{0,1\}$, $i = 1,2,\ldots,n$, where $n$ is the number of entities, in such a way that $x_i = 1$ (resp. $x_i = 0$) means that the entity $i$ is activated (resp. deactivated). However, the original definition of CA in [25] contemplates the possibility that the cells take state values in a finite set, although subsequently the majority of studies have been made in the case of Boolean CA. In fact, in experimental models, the state values of the entities can belong to a more general (finite) set. This situation naturally appears, for instance, when each entity can have different levels of activation or intensity, belonging to a totally ordered finite set which can be represented by $\{0,1,\ldots,m\}$ (see [24] for this approach in the context of probabilistic Boolean networks); or when each entity consists of several sub-entities, which can be activated or deactivated.

This last conception has inspired our extended model in this work. In this sense, we introduce a wide generalization of GDS, by considering that the state values of the entities can belong to an arbitrary Boolean algebra $B$ with $2^p$ elements, $p \in \mathbb{N}$, $p \geq 1$. This consideration widely extends the traditional one where it is assumed that every entity can take values only in the simplest Boolean algebra $\{0,1\}$.

In particular, we develop some techniques which allow us to study the orbit structure of these dynamical systems. As an application, we study the orbit structure of parallel dynamical systems over undirected graphs where the evolution operator is an arbitrary maxterm or minterm, using and generalizing at the same time the results in [1]. Moreover, taking into account the results in [3], we also analyze the case of parallel dynamical systems on general Boolean algebras whose evolution update scheme is defined by means of independent local functions chosen among $OR, AND, NAND$ and $NOR$. Finally, as a consequence, the results for parallel dynamical systems over directed dependency graphs in [2] and [3] can be also extended to this more general context.

This paper is organized as follows. In Section 2, we recall some notation concerning Boolean algebras and introduce the concepts of parallel and sequential dynamical systems on general Boolean algebras. Also in this second section, we determine the orbit structure of parallel dynamical systems on general Boolean algebras, whose evolution operator is given by the simplest maxterm $OR$ and the simplest minterm $AND$. In Section 3, we present a convenient adaptation of the Stones' representation theorem on which is based the posterior development that allows us to describe how the orbit structure of a parallel dynamical system on a general Boolean algebra is, either with a maxterm or minterm as evolution operator or with an evolution operator constituted by independent local functions.

## 2 Preliminaries and first results

Usually, in order to get a graphical idea of the situation, every entity is represented by a vertex of an undirected graph and two vertices are adjacent if their states influence each other in the update of the system. The undirected graph so built is called the *(undirected) dependency graph* of the system (see [10]).

If we denominate this graph $G = (V,E)$, where $V = \{1,2,\ldots,n\}$ is the vertex set and $E$ is the edge set, then, for each vertex/entity $1 \leq i \leq n$, it is natural to consider that its state $x_i \in \{0,1\}$. That is, the entity can be activated or deactivated.

On the other hand, for every vertex/entity $1 \leq i \leq n$, we need to consider all the vertices which influence it. Thus, we denote

$$A_G(i) = \{j \in V : \{j,i\} \in E\}$$

the sets of vertices which are adjacent to the vertex $i$.

Nevertheless, in many occasions, the influences are not bidirectional. This situation can be represented by an arc whose initial vertex is the influencing entity and the final vertex corresponds to the influenced entity, so obtaining a directed graph or digraph of relations. The directed graph so built is called the *directed dependency graph* of the system. In order to unify the notation in our results, it will be also denoted by $G = (V,E)$, although in this case $E$ is a set of arcs instead of edges. With the same aim, given a directed dependency graph and $i \in V$, $A_G(i)$ will stand for the set of vertices $j \in V$ such that there exists an arc from $j$ to $i$.

Observe that $(\{0,1\}, \vee, \wedge, ', 0, 1)$ is the simplest Boolean algebra, being

$$\begin{aligned} 1 \vee 0 = 0 \vee 1 = 1 \vee 1 = 1, \qquad & 0 \vee 0 = 0 \\ 1 \wedge 0 = 0 \wedge 1 = 0 \wedge 0 = 0, \qquad & 1 \wedge 1 = 1 \\ 1' = 0, \qquad & 0' = 1 \end{aligned}$$

Recall (see for instance [21]) that a Boolean algebra $(B, \curlyvee, \curlywedge, ', \mathbf{O}, \mathbf{I})$ is a bounded distributive lattice $(B, \curlyvee, \curlywedge)$

© 2015 NSP
Natural Sciences Publishing Cor.

with ordered structure given by

$$x \leq y \Longleftrightarrow x \curlyvee y = y \Longleftrightarrow x \curlywedge y = x \tag{1}$$

and such that

- $\curlyvee : B \times B \longrightarrow B$ is an inner operation in $B$ such that $x \curlyvee y = \sup\{x,y\}$, $x,y \in B$, when considering the ordered structure of $B$ as a lattice.
- $\curlywedge : B \times B \longrightarrow B$ is an inner operation in $B$ such that $x \curlywedge y = \inf\{x,y\}$, $x,y \in B$, when considering the ordered structure of $B$ as a lattice.
- $x' \in B$ stands for the complement of $x \in B$ (i.e. $x \curlyvee x' = \mathbf{I}$ and $x \curlywedge x' = \mathbf{O}$).
- $\mathbf{O} = \inf(B)$ satisfies that $x \curlyvee \mathbf{O} = x$ and $x \curlywedge \mathbf{O} = \mathbf{O}$, $x \in B$.
- $\mathbf{I} = \sup(B)$ satisfies that $x \curlyvee \mathbf{I} = \mathbf{I}$ and $x \curlywedge \mathbf{I} = x$, $x \in B$.

Here, we introduce a wide generalization of parallel dynamical systems over graphs, by considering that the states of the entities can take values in an arbitrary Boolean algebra $(B, \curlyvee, \curlywedge, ', \mathbf{O}, \mathbf{I})$ of $2^p$ elements, $p \in \mathbb{N}$, $p \geq 1$.

**Definition 2.1.** Let $G = (V,E)$ be a (directed or undirected) graph with $V = \{1,2,\ldots,n\}$. Then the following map

$$F : B^n \to B^n, \qquad F(x_1,x_2,\ldots,x_n) = (y_1,y_2,\ldots,y_n),$$

where $y_i$ is the updated state of the entity/vertex $i$ by locally applying the function $F$ over the states of the vertices in $\{i\} \cup A_G(i)$, constitutes a discrete dynamical system called a *parallel (discrete) dynamical system* (PDS) on the Boolean algebra $B$, over the dependency graph $G$ and with evolution operator $F$, which will be denoted by $[B,G,F]$.

In general, we will write $x_i^k$, to indicate the state value of the entity $i$ after $k$ iterations of the evolution operator $F$, while $x_i^0$ will stand for the initial state value of the entity $i$.

**Definition 2.2.** Let $G = (V,E)$ be a (directed or undirected) graph with $V = \{1,2,\ldots,n\}$ and $\pi = \pi_1\pi_2\cdots\pi_n$ a permutation on $V$. Then the following map

$$[F,\pi] = F_{\pi_n} \circ \cdots \circ F_{\pi_2} \circ F_{\pi_1} : B^n \to B^n,$$

where $F_{\pi_i} : B^n \to B^n$ is the update function on the state vector $(x_1,x_2,\ldots,x_n)$ which updates the state of the vertex $\pi_i$ while keeping the other states unchanged, constitutes a discrete dynamical system called a *sequential (discrete) dynamical system (SDS)* on the Boolean algebra $B$, over the dependency graph $G$ and with evolution operator $F$, which will be denoted by $[B,G,F,\pi]$.

Here, we mainly focus on the study of PDS over undirected graphs whose vertices/entities take values in an arbitrary Boolean algebra $B$ with $2^p$ elements, $p \in \mathbb{N}$, $p \geq 1$. As usual, we will assume that the graph is connected; otherwise one can work on each connected component of the graph analogously. Nevertheless, the kind of tools and reasonings which we develop in the next section can be easily adapted to study PDS over directed graphs and SDS over directed or undirected graphs.

The first question we deal with is the problem of determining the orbit structure of a PDS $[B,G,F]$ where $F = OR$ (resp. $F = AND$).

**Theorem 2.1.** Let $[B,G,OR]$ be the parallel dynamical system associated to the maxterm $OR$ over an undirected graph $G = (V,E)$, where the vertices/entities take values in a Boolean algebra $B$ with $2^p$ elements, $p \in \mathbb{N}$, $p \geq 1$. Then, all the orbits of this system are fixed points or eventually fixed points.

More precisely, the system presents exactly $2^p$ fixed points and the maximum number of iterations needed by an eventually fixed point to reach the corresponding fixed one is at most as large as the diameter d of the dependency graph.

**Proof.** It is not difficult to check that if all the vertices/entities have the same initial state value, i.e., $x_1^0 = x_2^0 = \cdots = x_n^0$, then they remain so forever. Since $x_i \in B$ for $i = 1,2,\ldots,n$, the possible initial state values for an entity are exactly $2^p$ and consequently this provides $2^p$ distinct fixed points of the system.

In general, denoting by

$$\alpha = \sup\{x_1^0,x_2^0,\ldots,x_n^0\},$$

we are able to prove that $x_i^k = \alpha$ for $i = 1,2,\ldots,n$ and $k \geq d$. In other words, the maximum number of iterations to attain an eventually fixed point is at most as large as the diameter of the dependency graph.

In order to do that, observe that given an entity $i$, for every entity $j$, $j \neq i$, there exists a path of length less than or equal to $d$ joining $i$ and $j$.

Then, thanks to the distributive property of the supremum, we have

$$x_i^d \geq \sup\{x_i^0,x_j^0\}$$

Since, this inequality yields for every $j \neq i$,

$$x_i^d \geq \alpha$$

On the other hand, it is clear that

$$x_i^k \leq \alpha$$

for every $k \in \mathbb{N}$.

Both inequalities allow us to infer that $x_i^d = \alpha$. As this reasoning can be done for $i = 1,\ldots,n$, after $d$ iterations all the vertices/entities have the same state value $\alpha$, i.e., $x_1^d = x_2^d = \cdots = x_n^d = \alpha$, and they remain so forever, giving as a result a fixed point of the system. □

Dually, we have

**Theorem 2.2.** Let $[B,G,AND]$ be the parallel dynamical system associated to the minterm $AND$ over an undirected graph $G = (V,E)$, where the vertices/entities take values in a Boolean algebra $B$ with $2^p$ elements, $p \in \mathbb{N}$, $p \geq 1$. Then,

© 2015 NSP
Natural Sciences Publishing Cor.

all the orbits of this system are fixed points or eventually fixed points.

More precisely, the system presents exactly $2^p$ fixed points and the maximum number of iterations needed by an eventually fixed point to reach the corresponding fixed one is at most as large as the diameter of the dependency graph.

Although these first results are quite intuitive, this intuitiveness disappears when we try to deal with the general problem of determining the orbit structure of a PDS $[B,G,F]$, where $F$ is an arbitrary maxterm or minterm. It motivates the development of the techniques which we introduce in the next section.

## 3 Main results

As was explained, the main purpose of this section is to describe the orbit structure of PDS over undirected graphs where the vertices/entities take values in a given Boolean algebra $(B,\curlyvee,\curlywedge,',\mathbf{O},\mathbf{I})$ with $2^p$ elements, $p\in\mathbb{N}$, $p\geq 1$.

More precisely, we will show how to translate the known results for the simplest case when $B=\{0,1\}$ to this more complex scenario. In this sense, the *Stone's Representation Theorem* for Boolean algebras (see, for instance, [21]) will play an essential role.

**Theorem 3.1.** Let $[B,G,F]$ be the parallel dynamical system associated to an evolution operator $F$ over an undirected graph $G=(V,E)$, where the vertices/entities take values in a Boolean algebra $B$ with $2^p$ elements, $p\in\mathbb{N}$, $p\geq 1$. Then, the state value of any entity $i$ can be represented by a Boolean state value vector $(x_{i1},x_{i2},\ldots,x_{ip})$, with $x_{ij}\in\{0,1\}$ for $j=1,2,\ldots,p$, such that the updating of the $j$-th coordinate only depends on the $j$-th coordinates of the state value vectors of the entities in the set $\{i\}\cup A_G(i)$.

**Proof.** Recall that, according to the ordered structure described in (1), the *atoms* of the Boolean algebra $B$ are the minimal elements of $B\setminus\{\mathbf{O}\}$. In particular, since $B$ has $2^p$ elements, $B$ has $p$ atoms. Denote by $A=\{a_1,a_2,\ldots,a_p\}$ the set of atoms of $B$, and by $\mathscr{P}(A)$ the power set (set of subsets) of $A$. As is well-known, $(\mathscr{P}(A),\cup,\cap,',\emptyset,A)$ is a Boolean algebra of $2^p$ elements (see, for instance, [21]).

Every element $x\in B\setminus\{\mathbf{O}\}$ can be univocally expressed as the disjunction (supremum) of a set of atoms

$$S_x=\{a_{x_1},a_{x_2},\ldots,a_{x_r}\}\in\mathscr{P}(A),\qquad 1\leq r\leq p,$$

where $r$ depends on $x$. That is,

$$x=a_{x_1}\curlyvee a_{x_2}\curlyvee\cdots\curlyvee a_{x_r}=\sup\{a_{x_1},a_{x_2},\ldots,a_{x_r}\}.$$

Then, the map

$$\Phi:B\longrightarrow\mathscr{P}(A)$$

defined by $\Phi(\mathbf{O})=\emptyset$ and $\Phi(x)=S_x$ is an isomorphism of Boolean algebras (Stone's Theorem).

Consider now the Boolean algebra $(\{0,1\}^p,\vee,\wedge,',\mathbf{0},\mathbf{1})$, whose elements are $p$-tuples $(x_1,x_2,\ldots,x_p)$, where $x_j\in\{0,1\}$ and,

- $(x_1,x_2,\ldots,x_p)\vee(y_1,y_2,\ldots,y_p)=(z_1,z_2,\ldots,z_p)$, $z_j=x_j\vee y_j$
- $(x_1,x_2,\cdots,x_p)\wedge(y_1,y_2,\cdots,y_p)=(z_1,z_2,\ldots,z_p)$, $z_j=x_j\wedge y_j$
- $(x_1,x_2,\ldots,x_n)'=(x_1',x_2',\ldots,x_p')$
- $\mathbf{0}=(0,0,\ldots,0)$
- $\mathbf{1}=(1,1,\ldots,1)$

Here, we are denoting in the same way the disjunction, conjunction and complement operators of the Boolean algebras $\{0,1\}$ and $\{0,1\}^p$, since they can be distinguished easily from the context.

An atom of $(\{0,1\}^p,\vee,\wedge,',\mathbf{0},\mathbf{1})$ is a $p$-tuple with 1 in one of the positions and 0's in the rest of positions. Then, reasoning as above, $\Psi:B\longrightarrow\{0,1\}^p$ defined by $\Psi(\mathbf{O})=\mathbf{0}$ and

$$\Psi(x)=(x_1,x_2,\ldots,x_p)\quad\text{with}\quad x_j=\begin{cases}1 & \text{if } a_{x_j}\in S_x\\ 0 & \text{if } a_{x_j}\notin S_x\end{cases}$$

is an isomorphism of Boolean algebras.

Bearing this in mind, let us take $[B,G,F]$ the PDS associated to the evolution operator $F:B^n\to B^n$, over the undirected graph $G=(V,E)$ whose vertices $V=\{1,2,\ldots,n\}$ take values in the Boolean algebra $B$ of $2^p$ elements. Then, without loss of generality, we can assume that the entities take values in $\{0,1\}^p$ by identifying each $x\in B$ with $\Psi(x)\in\{0,1\}^p$. That is, the state value of any entity $i$ can be represented by a Boolean state value vector $(x_{i1},x_{i2},\ldots,x_{ip})$, with $x_{ij}\in\{0,1\}$ for $j=1,2,\ldots,p$

Hence, we have to analyze how Boolean functions act over the Boolean algebra $\{0,1\}^p$. A Boolean function of $n$ variables over $\{0,1\}^p$ is a map

$$L:\{0,1\}^{pn}\longrightarrow\{0,1\}^p$$

where $L(X_1,X_2,\ldots,X_n)\in\{0,1\}^p$ is obtained from $X_1,X_2,\ldots,X_n\in\{0,1\}^p$ using the logical *AND*, the logical *OR*, the logical *NOT* and the elements $\mathbf{0},\mathbf{1}\in\{0,1\}^p$.

In particular, if $(X_1,X_2,\ldots,X_n)\in\{0,1\}^{pn}$ and $X_i=(x_{i1},x_{i2},\ldots x_{ip})\in\{0,1\}^p$

$$AND:\{0,1\}^{pn}\longrightarrow\{0,1\}^p,$$

is defined by $AND(X_1,\ldots,X_n)=(y_1,\ldots,y_p)$, where $y_j=AND(x_{1j},x_{2j},\ldots,x_{nj})$.

Likewise,

$$OR:\{0,1\}^{pn}\longrightarrow\{0,1\}^p$$

is given by $OR(X_1,\ldots,X_n)=(y_1,\ldots,y_p)$, where $y_j=OR(x_{1j},x_{2j},\ldots,x_{nj})$.

On the other hand,

$$':\{0,1\}^p\longrightarrow\{0,1\}^p$$

© 2015 NSP
Natural Sciences Publishing Cor.

is defined by $(x_1,x_2,\ldots,x_p)' = (x'_1,x'_2,\ldots,x'_p)$.

Therefore, since a Boolean evolution operator

$$F : \{0,1\}^{pn} \longrightarrow \{0,1\}^{pn}$$

of a PDS on the Boolean algebra $\{0,1\}^p$ consists of $n$ (local) Boolean functions,

$$L : \{0,1\}^{pn} \longrightarrow \{0,1\}^{p}$$

which act on an entity and its adjacent ones, it can be expressed by means of the Boolean functions *AND*, *OR* the logical *NOT* and the elements $\mathbf{0},\mathbf{1} \in \{0,1\}^p$. Moreover, from the argumentation above it follows that the updating of the $j$-th coordinate of a state value vector $(x_{i1},x_{i2},\ldots,x_{ip})$ of a vertex $i$, only depends on the $j$-th coordinates of the state value vectors of itself and its adjacent ones. □

In view of that, we can translate the results obtained in [1] to this much more general context as follows.

**Corollary 3.1.** Let $[B,G,MAX]$ be the parallel dynamical system associated to a maxterm *MAX* over an undirected graph $G = (V,E)$, where the vertices/entities take values in a Boolean algebra $B$. Then the periodic orbits of this system are fixed points or 2-periodic orbits.

As usual, dually we have,

**Corollary 3.2.** Let $[B,G,MIN]$ be the parallel dynamical system associated to a minterm *MIN* over an undirected graph $G = (V,E)$, where the vertices/entities take values in a Boolean algebra $B$. Then the periodic orbits of this system are fixed points or 2-periodic orbits.

In [3], the cases concerned with parallel dynamical systems with *OR*, *AND*, *NAND* and *NOR* functions as independent local functions were analyzed. Taking our previous reasonings into account, it can be stated the following corollary.

**Corollary 3.3.** Let $[B,G,\{f_i\}]$ a PDS associated to the local Boolean functions $f_i \in \{AND,OR,NOR,NAND\}$ over an undirected graph $G = (V,E)$, where the vertices/entities take values in a Boolean algebra $B$. Then the periodic orbits of this system are fixed points or 2-periodic orbits.

In a similar way, we can extend the results for PDDS shown in [2] and [3] to this more general context.

## 4 Conclusions

We provide a method that allows us to study the orbit structure of GDS, for which the states of the entities can take values in an arbitrary Boolean algebra with $2^p$ elements, $p \in \mathbb{N}$, $p \geq 1$.

In fact, since any set with structure of Boolean algebra is isomorphic to a subalgebra of a Boolean algebra of the form $\{0,1\}^p$ and then one could establish a injection from this algebra to $\{0,1\}^p$ in order to make a correspondence among elements of the Boolean algebra and elements of $\{0,1\}^p$, we also solve the problem of studying systems where the entities can take state values in different (finite) Boolean algebras.

In relation to previous studies in this direction on BN [24], we study completely the problem when the state values set is any Boolean algebra in the more general context of PDS.

In particular, we analyze the orbit structure of PDS over undirected graphs. Nevertheless, the kinds of tools and reasonings which we develop in this work can be used to study PDS over directed graphs and SDS over directed or undirected graphs.

This also constitutes an important issue for applications, since it provides an appropriate model for the case where entities of a system are composed of a finite number of parts that can be activated or deactivated.

## Acknowledgement

This work has been partially supported by the Spanish national grant MTM2011-23221 and by the grant PEII-2014-001-A.

## References

[1] J.A. Aledo, S. Martinez, F.L. Pelayo and J.C. Valverde, Mathematical and Computer Modelling **35**, 666-671 (2012).
[2] J.A. Aledo, S. Martinez and J.C. Valverde, Applied Mathematics and Computation **219**, 1114-1119 (2012).
[3] J.A. Aledo, S. Martinez and J.C. Valverde, Journal of Computational and Applied Mathematics **237**, 335-339 (2013).
[4] J.A. Aledo, S. Martinez, J.C. Valverde, Int. J. Comput. Math. **90(9)**, 1796–1808 (2013).
[5] J.A. Aledo, S. Martinez, J.C. Valverde, Int. J. Comput. Math. **90(10)**, 2039-2048 (2013).
[6] C.L. Barret, C.M. Reidys, Appl. Math. Comput. 98 (1999) 241-259.
[7] C.L. Barret, H.S. Mortveit, C.M. Reidys, Applied Mathematics and Computation **107**, 121-136 (2002).
[8] C.L. Barret, H.S. Mortveit, C.M. Reidys, Applied Mathematics and Computation **122**, 325-340 (2002).
[9] C.L. Barrett, H.S. Mortveit, C.M. Reidys, Applied Mathematics and Computation **134**, 153-171 (2003).
[10] C.L. Barret, W.Y.C. Chen, M.J. Zheng, Mathematics and Computers in Simulation **66**, 487-497 (2004).
[11] H.J. Blok, B. Bergersen, Physical Review E **59**, 3876-3879 (1999).
[12] C. Bisi, G. Chiaselotti, Advances in Geometry **3**, 1–27 (2013).
[13] ST Dougherty, B Yldz, S Karadeniz, Self-Dual Codes over R_k and Binary Self-Dual Codes, European Journal of Pure and Applied Mathematics, **6**, (2013).
[14] G. Chiaselotti, G.Marino, P. A. Oliverio, D. Petrassi, Journal of Applied Mathematics **1**, Article ID 973501, 10 pages (2013).
[15] J. Kari, Theoretical Computer Science **334**, 3-33 (2005).

© 2015 NSP
Natural Sciences Publishing Cor.

[16] S. A. Kauffman, Journal of Theoretical Biology **22**, 437-467 (1969).
[17] S. A. Kauffman, Oxford University Press, Oxford, 1993.
[18] G Beer, A Caserta, G Di Maio, R Lucchetti, Convergence of partial maps, Journal of Mathematical Analysis and Applications, **419**, 1274-1289 (2014). Elsevier
[19] T. Mihaela, T. Matache, J. Heidel, Physical Review E **71**, 1-13 (2005).
[20] H.S. Mortveit, C.M. Reidys, An Introduction to Sequential Dynamical Systems, Springer, New York, 2007.
[21] K. Rosen, Discrete Mathematics and Its Applications, McGraw-Hill Education, New York, 2011.
[22] J.L. Schiff, Cellular automata: a discrete view of the world, Wiley, New York, 2008.
[23] B. Schönfisch, A. de Roos, BioSystems **51**, 123-143 (1999).
[24] R.V. Sole, B. Luque, S.A. Kauffman, Phase transitions in random networks with multiple states, Technical Report 00-02-011, Santa Fe Institute, 2000.
[25] S. Wolfram, Reviews of Modern Physics **55**, 601-644 (1983).
[26] S. Wolfram, Cellular Automata and Complexity, Addison-Wesley, New York, 1994.

**Juan A. Aledo** graduated from the University of Murcia (Spain) and earned a PhD degree from the doctoral programme of Mathematics at the same University. He is a member of the Institute of Mathematical Research Applied to Science and Engineering of Castilla-La Mancha and a Professor of the Department of Mathematics at the University of Castilla-La Mancha (Spain). His research interests include differential geometry and discrete mathematics in relation to computer science. He has published more than 40 journal articles.

**Silvia Martinez** graduated from the Polytechnic University of Valencia (Spain) and earned a PhD degree from the doctoral programme of Physics and Mathematics at the University of Castilla-La Mancha (Spain). She is a member of the Institute of Mathematical Research Applied to Science and Engineering of Castilla-La Mancha and a Professor of the Department of Mathematics at the University of Castilla-La Mancha. Her research interests are in the areas of applied mathematics and mathematical physics including combinatorial dynamics, discrete mathematics in relation to computer science, application of graph theory to circuits, networks and more generally to information and communication. She has been visiting Professor at the University of Calabria (Italy). She has published several research articles and has also been referee in some reputed international journals of mathematical and computational sciences.

**Jose C. Valverde** graduated from the University of Murcia (Spain) and earned a PhD degree from the doctoral programme of Mathematics at the same University. He is a member of the Institute of Mathematical Research Applied to Science and Engineering of Castilla-La Mancha and a Professor of the Department of Mathematics at the University of Castilla-La Mancha (Spain). His research interests include dynamical systems, biomathematics and discrete mathematics in relation to computer science. He is the author/coauthor of 5 books and has published more than 30 journal articles. He has served as editor-in-chief, chief editor, guest editor and editorial board member of several indexed international journals and has referred papers for several JCR-journals. He has been visiting Professor at some Universities and main speaker in some international conferences. He has also participated as a scientific committee member or as a technical expert committee member of some international congresses.

© 2015 NSP
Natural Sciences Publishing Cor.