3-2006

Asynchronous Random Boolean Network Model with Variable Number of Parents based on Elementary Cellular Automata Rule 126

Mihaela Teodora Matache
University of Nebraska at Omaha, dvelcsov@unomaha.edu

Asynchronous Random Boolean Network Model with Variable Number of Parents based on Elementary Cellular Automata Rule 126

Mihaela T. Matache

Department of Mathematics
University of Nebraska at Omaha
Omaha, NE 68182-0243, USA
dmatache@mail.unomaha.edu

Abstract A Boolean network with N nodes, each node's state at time t being determined by a certain number of parent nodes, which can vary from one node to another is considered. This is a generalization of previous results obtained for a constant number of parent nodes, by Matache and Heidel in *Asynchronous random Boolean network model based on elementary cellular automata rule 126*, Phys. Rev. E 71, 026232, 2005. The nodes, with randomly assigned neighborhoods, are updated based on various asynchronous schemes. The Boolean rule is a generalization of rule 126 of elementary cellular automata, and is assumed to be the same for all the nodes. We provide a model for the probability of finding a node in state 1 at a time t for the class of generalized asynchronous random Boolean networks (GARBN) in which a random number of nodes can be updated at each time point. We generate consecutive states of the network for both the real system and the models under the various schemes, and use simulation algorithms to show that the results match well. We use the model to study the dynamics of the system through sensitivity of the orbits to initial values, bifurcation diagrams, and fixed point analysis. We show that the GARBN's dynamics range from order to chaos depending on the type of random variable generating the asynchrony and the parameter combinations.

1. Introduction

Boolean network models have been used extensively in modelling networks in which the node activity can be described by two states, ON and OFF, "active and nonactive", "responsive and nonresponsive", "upregulated and downregulated", and in which each node is updated based on logical relationships with other nodes, called parents. Although the Boolean network models may simplify the reality, they retain meaningful information regarding the dynamics of the system. Moreover, these models are easy to understand and use.

Applications of Boolean networks have started with the work of Kauffman ([1] - [3]). The dynamics of "spin-glasses" have been influential in the formulation of Kauffman's N/K models used in his random

Boolean networks and other complex, adaptive systems [3]. Such models have applications in solid-state physics and condensed matter, where it is important to study the dynamics of interactive systems through the changing of connectivity rules and analysis of the resulting phenomena, with the goal of understanding how the system can be moved into and out of equilibrium states. At the same time, in statistical mechanics, it is important to understand the dynamics of large interacting systems where the nodes are randomly connected to other nodes and have various functions [4].

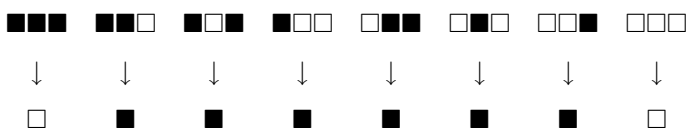
Other applications of Boolean and random Boolean networks include biochemical and genetic networks ([5] - [17]) where identifying a possible steady-state behavior of a tumor can give insight into how therapists may develop treatments to alter and cure it, or models for random interaction between two-state neurons in a neural network [18].

Recently, Matache and Heidel [19] have considered a simple asynchronous Boolean network with N nodes, each node being influenced by exactly k other nodes (parents). The number of parents is considered fixed for all nodes. The Boolean rule is a generalization of rule 126 of elementary cellular automata (ECA). They use various updating schemes to generate the asynchrony in the network. In particular they consider the case of Asynchronous Random Boolean Networks (ARBN) in which only one node is updated at every time step, and the class of Generalized ARBNs (GARBN) in which a random number of nodes can be updated at each time point. We note that this classification of asynchronous Boolean networks is due to Gershenson [20]. The asynchrony for GARBNs is generated using various random variables. They show, both theoretically and by example, that the ARBNs generate an ordered behavior regardless of the particular updating order used, whereas the GARBNs have behaviors that range from order to chaos depending on the type of random variable used to determine the number of nodes to be updated and the parameter combinations. However, high connectivity swamps out chaos and periodicity and leaves only stable fixed points. This regularity is shown to be quite general with exceptions occurring only for a small number of neighborhood distributions.

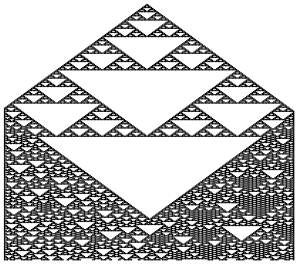
In this paper we extend that work by allowing a nonconstant number of parents for the nodes of the network, and assume the same generalization of ECA rule 126. In [21] the same authors have provided such an extension for a synchronous random Boolean network governed by the same generalization of rule 126 of cellular automata, showing that the route to chaos is due to a cascade of period doubling bifurcations which turn into reversed bifurcations for various combinations of the parameters. It is shown that high connectivity leads to order.

We note that although most of the Boolean network models in the literature assume a synchronous updating of the nodes, asynchronous models are more plausible for many biological phenomena or cellular automata. For example, in [22], Cornforth observes that individual ants display aperiodic patterns of active and resting periods, while the colony as a whole may exhibit synchronized activity. At the same time asynchronous activity of the neurons in the brain could lead to some global patterns.

Rule 126 of elementary cellular automata can be described as follows



where black is ON and white is OFF. Rule 126 is a "legalistic" and "totalistic" ECA rule ([21], [23]), which exhibits a complex behavior despite its simplicity. It provides a good model for cell growth and for chemical catalytic processes where the central site survives (or is born) unless the neighborhood is completely non-populated or completely crowded, in which case it dies [19]. Cellular automata provide good models in biological and physical systems generating patterns ([24], [25]). To see how rule 126 can generate complex patterns we present below the pattern formed when starting with only one black (ON) node in an elementary cellular automaton of 200 nodes, and iterating the system 250 time steps (the time increases downward in the figure).



In Section 2 we show that the formula for the probability of finding a node in state 1 for the case of GARBNs with variable number of parent nodes, and Boolean rule given by the ECA rule 126, is

$$p(t+1) = \sum_{j=1}^{J} \frac{M_j}{N} \left\{ \frac{N_1^{k_j}(t)}{M_j} \left[1 - \frac{x_t}{N} \left(1 - (1-p(t))^{k_j} + p(t)^{k_j} \right) \right] + \frac{x_t}{N} (1 - (1-p(t))^{k_j} \right\}$$

where N is the size of the network, $k_1, k_2, \ldots, k_J$ the distinct values for the number of parents of the nodes, M_j the number of nodes with k_j parents, $N_1^{k_j}(t)$ the number of nodes with k_j parents that are in state 1 at time t for $j = 1, 2, \ldots, J$, and x_t the number of nodes to be updated at time t. We provide a simulation algorithm that allows us to generate consecutive states of the model and match the results with iterations of the real system.

In Section 3 we study the dynamics of the system through the analysis of the sensitivity of the orbits to the initial values, bifurcation diagrams, and fixed points. We use various random distributions to generate the number x_t. We show that the system may exhibit order or chaos, depending on the underlying parameters, the distributions used, and the number of nodes to be updated at each time point. The route to chaos is due to period-doubling bifurcations which turn into reversed bifurcations for certain combinations of parameter values. The findings are explained in more detail together with a thorough analysis of the two and three dimensional cases in Section 4 which focuses on bifurcations. We

show that in general, when few nodes are updated at the same time, the system exhibits order, while for the case when a large number of nodes are updated at each time point, the system could exhibit chaos which may be reversed for high values of the connectivity parameters.

Section 5 is dedicated to conclusions and further directions of investigation.

2. The Random Boolean Network Model

In this section we describe the proposed model for the Boolean network under consideration. We show how this model compares to those in [19] and [21].

We consider a network with N nodes, $c_1, c_2, \dots c_N$, that can take on two values, 1 or 0. At each time point t the system can be in one of the 2^N possible states. The nodes are updated in an asynchronous fashion from time t to time $t+1$ according to a generalized ECA rule 126. Each node c_n is assigned a random "neighborhood" of parents, which may vary in size from one node to another. If a node has a "neighborhood" of size k, then the k parents are selected randomly from the remaining $N-1$ nodes (with probability $1/\binom{N-1}{k}$). The Boolean rule can be described as follows: if a node c_n and all its parents have the same value at time t (that is they are all either 0 or 1), then $c_n(t+1) = 0$, otherwise $c_n(t+1) = 1$. This generalizes rule 126 of cellular automata ([21], [23]).

The system is basically described by the number of parents of each node. The quantity $N_1(t) := \sum_{n=1}^{N} c_n(t)$ gives the number of cells that are in state 1 at time t. The concentration of nodes in state 1 is given by $\frac{1}{N}\sum_{n=1}^{N} c_n(t)$, and will be used to estimate the probability $p(t+1)$ that a node is in state 1 at time $t+1$, given $p(t)$. The formula we obtain will help us study the dynamics of the system in Section 3.

In the paper [19] the authors show that $p(t+1)$ is given by

$$p(t+1) = p(t) + \frac{1}{N}\left[1 - p(t) - p(t)^{k+1} - (1-p(t))^{k+1}\right]. \tag{1}$$

for the ARBN case when only one node is updated at each time point, and

$$p(t+1) = p(t) + \frac{x_t}{N}\left(1 - p(t) - (1-p(t))^{k+1} - p(t)^{k+1}\right). \tag{2}$$

for the GARBN case when a random number of nodes is updated at each time point. In these formulae $k \geq 1$ is the number of parents of each node (considered fixed), and x_t is the number of nodes to be updated at time t (randomly generated).

In the paper [21] the formula for the probability $p(t+1)$ that a node is in state 1 at time $t+1$ given $p(t)$ is

$$p(t+1) = \sum_{j=1}^{J} \frac{M_j}{N}\left[1 - \frac{N_0^{k_j}(t)}{M_j}(1-p(t))^{k_j} - \frac{N_1^{k_j}(t)}{M_j}p(t)^{k_j}\right]$$

where N is the size of the network, $k_1, k_2, \dots, k_J$ the distinct values for the number of parents of the nodes, M_j the number of nodes with k_j parents, $N_0^{k_j}(t)$ the number of nodes with k_j parents that are

in state 0 at time t, and $N_1^{k_j}(t)$ the number of nodes with k_j parents that are in state 1 at time t. The network is considered synchronous.

Our goal is to provide a similar formula when dealing with nonconstant number of parents and asynchronous updating at the same time. We follow the notations used in [21]. Let $k_1, k_2, \ldots, k_J$ be the distinct values for the number of parents the nodes $c_1, c_2, \ldots, c_N$ can have. Also let C_j be the collection of all nodes having k_j parents, and M_j be the number of nodes in each class C_j, $j = 1, 2, \ldots, J$. Also, let $N_0^{k_j}(t)$ be number of nodes of class C_j in state 0 at time t, and $N_1^{k_j}(t)$ the number of nodes of class C_j in state 1 at time t, $j = 1, 2, \ldots, J$. It follows that $\sum_{j=1}^{J}(N_0^{k_j}(t) + N_1^{k_j}(t)) = N$, and $N_0^{k_j} + N_1^{k_j} = M_j$, $j = 1, 2, \ldots, J$. The probability that a node is in state 1 at time t is given by $p(t) = \frac{1}{N}\sum_{j=1}^{J} N_1^{k_j}(t)$. We want to compute the conditional probability that a node is in state 1 at $t+1$, given the known probability $p(t)$. Observe that this is determined by the number of nodes that change from state 0 at time t to state 1 at time $t+1$ and the number of nodes that remain in state 1 from time t to $t+1$.

If x_t is the number of nodes to be updated at time t, denote by x_t^j the number of nodes in class C_j that are updated at time t. It follows that $\sum_{j=1}^{J} x_t^j = x_t$. Also, let $Nx_0^{k_j}$ and $Nx_1^{k_j}$ be the number of nodes of class C_j that are 0 and 1, respectively, and are updated at time t. Thus $Nx_0^{k_j} + Nx_1^{k_j} = x_t^j$.

We will start with the derivation of $N_{0\to1}^{k_j}(t)$ which will denote the number of nodes of class C_j that are 0 at time t and become 1 at time $t+1$. We use the notation $\mathcal{P}$ for the probability of an event and $p(t)$ for the probability of a node being in state 1 at time t. If $c_n(t) = 0$ and the node has k parents, then

$$\mathcal{P}(c_n(t+1) = 1 | c_n(t) = 0) = \mathcal{P}(\text{at least one of the parents of node } c_n \text{ is 1 at time } t) =$$

$$= 1 - \mathcal{P}(\text{all parents of node } c_n \text{ are 0 at time } t) = 1 - (1 - p(t))^k.$$

Here k denotes the number of parents of the node c_n and could be any of the numbers $k_1, k_2, \ldots, k_J$. We assume that the parents can be in state 0 or 1 independently of each other. At time $t+1$ we could have $0, 1, 2, \ldots,$ or $Nx_0^{k_j}(t)$ nodes going from state 0 at time t to state 1 at $t+1$. We define the discrete random variable X given by the probability distribution function

$$\mathcal{P}(X = l) = \mathcal{P}(\text{l nodes of class } C_j \text{ go from state 0 at time } t \text{ to state 1 at time } t+1) =$$

$$= \binom{Nx_0^{k_j}(t)}{l} \left[1 - (1 - p(t))^{k_j}\right]^l \left[(1 - p(t))^{k_j}\right]^{Nx_0^{k_j}(t) - l}, \qquad l = 0, 1, 2, \ldots, Nx_0^{k_j}(t).$$

One can check by a straightforward computation that $\sum_{l=0}^{Nx_0^{k_j}(t)} \mathcal{P}(X = l) = 1$. Then $N_{0\to1}^{k_j}(t)$ will be the expected value of X, that is

$$N_{0\to1}^{k_j}(t) = \sum_{l=0}^{Nx_0^{k_j}(t)} l\,\mathcal{P}(X = l) = Nx_0^{k_j}(t)\left[1 - (1 - p(t))^{k_j}\right].$$

Note that if the number of parents is the same for all nodes, say k, then $k_j = k$, for all $j = 1, 2, \ldots, J$ and the total number of nodes going from state 0 at time t to state 1 at time $t+1$ is given by

$$N_{0\to 1}(t) = Nx_0(t)\left(1-(1-p(t))^k\right), \qquad Nx_0(t) = \sum_{j=1}^{J} Nx_0^{k_j}(t).$$

This represents exactly the formula obtained in [19] for $N_{0\to 1}(t)$, where $Nx_0(t)$ is expressed as $x_t(1-p(t))$.

By a similar argument, one can write the following formulas for $N_{1\to 1}^{k_j}(t)$, the number of nodes of class C_j that remain 1 from time t to $t+1$, $N_{0\to 0}^{k_j}(t)$ the number of nodes of class C_j that remain 0, and $N_{1\to 0}^{k_j}(t)$, the number of nodes of class C_j that change from 1 to 0. In each case an appropriate random variable is defined as in the case of $N_{0\to 1}^{k_j}(t)$, and the number of nodes going from one state at time t to the next state at time $t+1$ is defined as the expected value of that random variable. Thus we obtain the following:

$$N_{1\to 1}^{k_j}(t) = (N_1^{k_j}(t) - Nx_1^{k_j}(t)) + Nx_1^{k_j}(t)(1-p(t)^{k_j}),$$

$$N_{0\to 0}^{k_j}(t) = (N_0^{k_j}(t) - Nx_0^{k_j}(t)) + Nx_0^{k_j}(t)(1-p(t))^{k_j},$$

$$N_{1\to 0}^{k_j}(t) = Nx_1^{k_j}(t)p(t)^{k_j}.$$

Again, by setting all numbers k_j equal to k and performing the computations we get the formulas obtained in [19], namely

$$N_{1\to 1}(t) = N_1(t) - Nx_1(t)p(t)^k, \quad N_{0\to 0}(t) = (N_0(t) - Nx_0(t)) + Nx_0(t)(1-p(t))^k, \quad N_{1\to 0}(t) = Nx_1(t)p(t)^k$$

where $N_0(t) = \sum_{j=1}^{J} N_0^{k_j}(t)$, $N_1(t) = \sum_{j=1}^{J} N_1^{k_j}(t)$, $Nx_0(t) = \sum_{j=1}^{J} Nx_0^{k_j}(t)$, $Nx_1(t) = \sum_{j=1}^{J} Nx_1^{k_j}(t)$. In [19] $Nx_0(t) = x_t(1-p(t))$, $Nx_1(t) = x_t p(t)$.

It follows immediately that the sum of all these quantities is equal to N. We can now construct the quantities $p_j(t+1) = \frac{1}{N}\left[N_{0\to 1}^{k_j}(t) + N_{1\to 1}^{k_j}(t)\right]$ where $j = 1, 2, \ldots, J$, representing the probabilities of finding a node of class C_j in state 1 at time $t+1$. Observe that after a short computation

$$p_j(t+1) = \frac{n_1^{k_j}(t)}{N} + \frac{Nx_0^{k_j}(t)}{N}(1-(1-p(t))^{k_j}) - \frac{Nx_1^{k_j}(t)}{N}p(t)^{k_j}.$$

Now $Nx_1^{k_j}(t)$ can be expressed as a proportion of x_t as follows $Nx_1^{k_j}(t) = \frac{N_1^{k_j}(t)}{N}x_t$, and consequently $Nx_0^{k_j}(t) = \frac{M_j - N_1^{k_j}(t)}{N}x_t$. Thus, after replacing these in the expression of $p_j(t+1)$ and performing some computations we get

$$p_j(t+1) = \frac{M_j}{N}\left[\frac{N_1^{k_j}(t)}{M_j}\left(1 - \frac{x_t}{N}\left(1-(1-p(t))^{k_j} + p(t)^{k_j}\right)\right) + \frac{x_t}{N}\left(1-(1-p(t))^{k_j}\right)\right].$$

The quantity $\frac{N_1^{k_j}(t)}{M_j}$ represents the proportion of nodes of class C_j that are 1 at time t. Thus we can write the final formula for the probability that a node is in state 1 at time $t+1$, by summing up the

$p_j(t+1)$ for all $j = 1, 2, \ldots, J$ as follows

$$p(t+1) = \sum_{j=1}^{J} \frac{M_j}{N} \left[\frac{N_1^{k_j}(t)}{M_j} \left(1 - \frac{x_t}{N} \left(1 - (1 - p(t))^{k_j} + p(t)^{k_j} \right) \right) + \frac{x_t}{N} \left(1 - (1 - p(t))^{k_j} \right) \right] \tag{2.1}$$

We include here the formula for $p(t+1)$ obtained in [21] for a synchronous network with multiple values of the number of parents

$$p(t+1) = \sum_{j=1}^{J} p_j(t+1) = \sum_{j=1}^{J} \frac{M_j}{N} \left[1 - \frac{N_0^{k_j}(t)}{M_j} (1 - p(t))^{k_j} - \frac{N_1^{k_j}(t)}{M_j} p(t)^{k_j} \right].$$

Observe the similarities when replacing $N_0^{k_j}(t)$ by $M_j - N_1^{k_j}(t)$. We also include the formula for $p(t+1)$ obtained in [19] for a GARBN with a constant number of parents k

$$p(t+1) = \frac{N_1(t+1)}{N} = p(t) + \frac{x_t}{N} \left[1 - p(t) - p(t)^{k+1} - (1 - p(t))^{k+1} \right].$$

Again we can observe the similarities if we collect together the terms involving $\frac{x_t}{N}$ in formula (2.1).

Note that if all the nodes are 0 at time t, then $N_0^{k_j}(t) = N, N_1^{k_j}(t) = 0$, for all $j = 1, 2, \ldots J$, so that $p(t+1) = 0$, which is to be expected since by the Boolean rule all the nodes stay 0 at time $t+1$. Similarly, if all the nodes are 1 at time t, $p(t+1) = \frac{N - x_t}{N}$ by the formula, as well as by the Boolean rule.

Given all of the above, we propose the following simulation algorithm for the Boolean network under consideration. The algorithm provides the computation of $p(t)$ for all $t = 0, 1, 2, \ldots$.

- For $t = 0$ choose arbitrary numbers $p_j(0) \in [0, \frac{M_j}{N}]$, $j = 1, 2, \ldots, J$, and let

$$p(0) = \sum_{j=1}^{J} p_j(0).$$

- For each $t = 0, 1, 2, \ldots$ compute

$$p_j(t+1) = \frac{M_j}{N} \left[\frac{p_j(t)}{\frac{M_j}{N}} \left(1 - \frac{x_t}{N} \left(1 - (1 - p(t))^{k_j} + p(t)^{k_j} \right) \right) + \frac{x_t}{N} \left(1 - (1 - p(t))^{k_j} \right) \right] \tag{2.2}$$

  where $j = 1, 2, \ldots, J$, and let

$$p(t+1) = \sum_{j=1}^{J} p_j(t+1).$$

The formula for $p_j(t+1)$ in (2.2) is similar to the summands for $p_j(t+1)$ in (2.1).

We can use this algorithm to simulate consecutive states of the model and compare the results with iterations of the real system. In this paper we present only typical graphs, as the result of numerous simulations performed in Matlab. The asynchrony is generated using various random distributions: discrete uniform on $\{1, 2, \ldots, N\}$, binomial with N trials and probability of success θ, Poisson with parameter λ, power law with parameter γ, and hypergeometric with N balls, w white balls, $N - w$ black balls, and s selected balls.

The graphs in Figures 1-2 represent simulations of the (2-dimensional) model and the actual Boolean system for some parameter choices and the binomial distribution for x_t. The model is represented by the mesh, while the real system is represented by points. We graph $p(t+\text{iteration})$ versus $p_1(t)$ and $p_2(t)$ for the iteration specified in each graph. We can deduce the behavior of the system and the model for other cases from these graphs, since all the other simulations obtained for various parameter combinations are quite similar to those in Figures 1-2. We have simulated similar graphs for the following distributions of x_t: uniform, Poisson, binomial, power law, hypergeometric. In all cases the parameters of the distributions are chosen such that the probability of a value x_t larger than N is practically zero. In Figure 1 we observe that although the match between the model and the real system is not perfect in the beginning, in the long run they match quite well. Figure 1 is representative for almost all the cases considered. The only situation for which the situation differs is the binomial distribution with N trials and a large probability of success. In this case both the system and the model do not settle even after a significantly large number of iterations as seen in Figure 2.

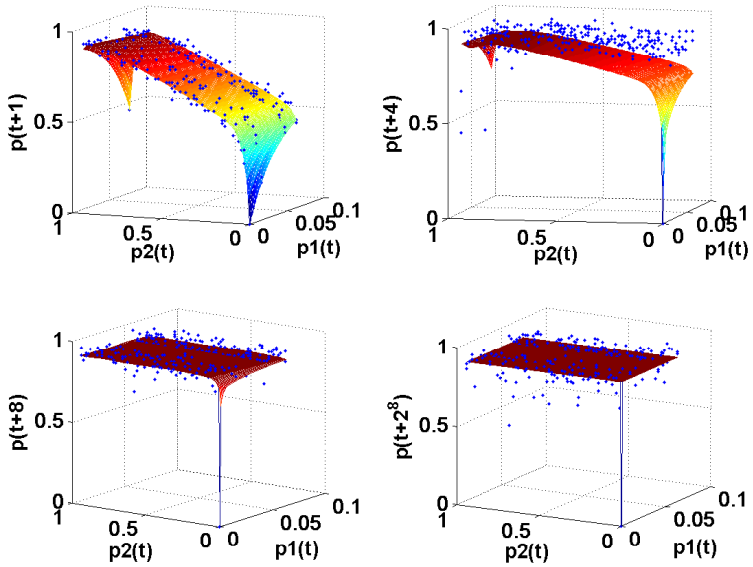


FIGURE 1. Iterations of the system and the model, with $N = 64, k_1 = 4, k_2 = 32, M_1 = 4, M_2 = 60$, and x_t from a binomial distribution with number of trials N and probability of success $\theta = 0.5$. The model is a good approximation for the system after a transient period. This figure is similar to other graphs obtained with other distributions such as uniform, Poisson, power law, and hypergeometric.

3. System Dynamics

Now we can study the dynamics of the system by analyzing the model and its behavior. We start with the study of the stability of the orbits to initial conditions. We consider the case of only two distinct

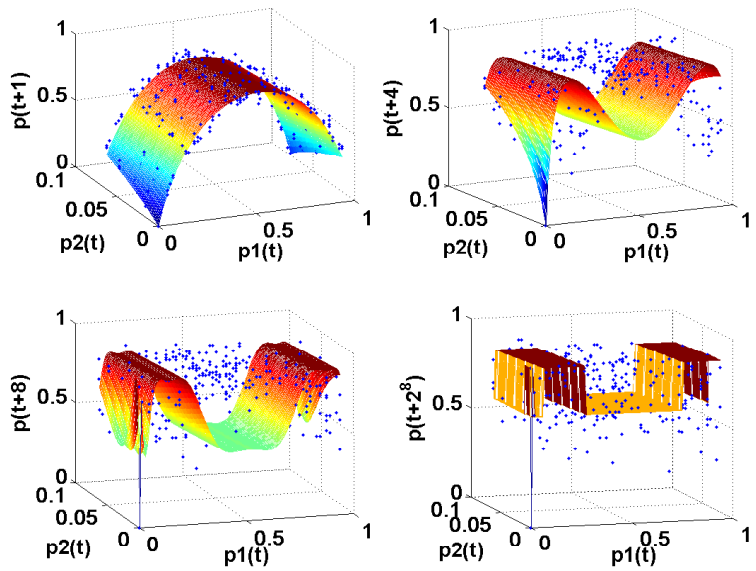


FIGURE 2. Iterations of the system and the model, with $N = 64, k_1 = 4, k_2 = 32, M_1 = 60, M_2 = 4$, and x_t from a binomial distribution with number of trials N and probability of success $\theta = 0.9$. We observe that both the model and the real system do not settle even after many iterations.

values for the number of parents for simplicity. We fix the parameters N, M_1, M_2, k_1, k_2, and generate x_t according to various distributions: uniform, binomial, Poisson, power law, and hypergeometric. We choose two initial pairs $(p_1(0), p_2(0))$ and $(q_1(0), q_2(0))$ as starting points for the orbits. We iterate many times the equations of the model and compute $p(t) = p_1(t) + p_2(t)$ and $q(t) = q_1(t) + q_2(t)$ for each time point t. Then we plot the error $E(t) = |p(t) - q(t)|$ versus t. In Figure 3 we show the case of $N = 1024, M_1 = M_2 = \frac{N}{2}, k_1 = 4, k_2 = 128$ and x_t from a power law distribution with parameter $\gamma = 2$. In this graph $p(0) = 0.375$ and $q(0) = 0.415$, so the starting values are relatively close. This graph is typical and very similar for most other combinations of parameters considered in the experiments. We observe that the error converges to zero. The rates of convergence to zero may vary based on the closeness of the initial values and the parameter combinations. Thus the system does not exhibit sensitivity to initial values and this is consistent with the ordered behavior observed in the bifurcation diagrams which will be analyzed next.

The case when x_t is generated with a binomial distribution exhibits a more complicated behavior. For example, in Figure 4 we present the case of $N = 2^{10}, M_1 = N/16, k_1 = 50, k_2 = 256$, and x_t from a binomial distribution with N trials and probability of success 0.9. We observe that the error does not converge to zero even after $10,000$ iterations. However, for values of k_2 smaller than approximately 45 or larger than approximately 340 (not shown but similar to Figure 3) the error converges to zero rather fast. This suggests a range of values of k_2 for which the system may exhibit chaos, which is reversed

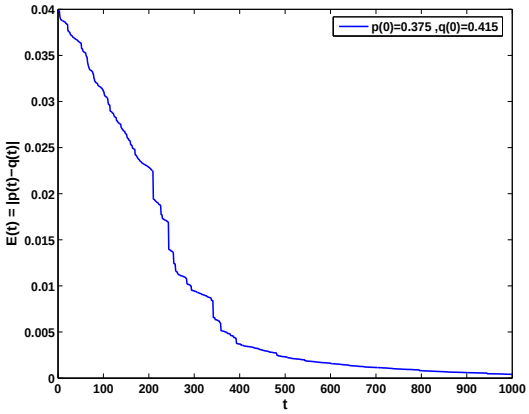


FIGURE 3. Error plot for the ARBN model with $N = 1024, M1 = 512, M2 = 512, k_1 = 4, k_2 = 128$, and x_t from a power law distribution with parameter $\gamma = 2$. This graph is typical for all the distributions considered by the authors, with various parameters, except the binomial distribution with a large probability of success, and hypergeometric with large values of the parameters. We note that both these situations correspond to the case when the generated values of x_t are rather large.

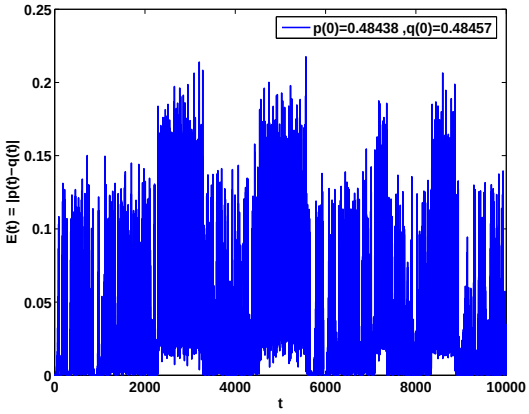


FIGURE 4. Error plot for the ARBN model with $N = 1024, M1 = 64, M2 = 960, k_1 = 50, k_2 = 256$, and x_t from a binomial distribution with N trials and probability of success 0.9. We observe a clear sensitivity to initial values.

for large values of k_2. A similar behavior is observed for other parameter combinations with the note that the values of k_1 and k_2 for which sensitivity of the orbits to initial values is observed, decrease with the increase of the probability of success of the binomial distribution. Similar situations occur for the uniform and the Poisson distributions. The bifurcation diagrams will clarify this situation.

Finally, in the case when x_t is generated with a hypergeometric distribution with the following parameters: N the total number of balls (equal to the total number of nodes in the network), w the number of white balls, and s the number of selected balls, the error may or may not converge to zero, depending

on the parameter combinations. More precisely, it is observed that if the parameters w and s are such that the values x_t are not very large, the error converges to zero. The same behavior is observed for cases when the values x_t are large, but the connectivity values k_1 and k_2 are both small enough. This behavior is basically independent of the proportions $\frac{M_1}{N}, \frac{M_2}{N}$ as long as the parameters k_1 and k_2 are small enough. When the parameters are large enough, the error does not converge to zero. To provide a better graphical view of the situation, we employ some three dimensional graphs as follows. In Figure 5 we consider the following fixed parameters: $N = 2^{10}, M_1 = 960, M_2 = 64, w = 900, s = 700$. We allow k_1 and k_2 to move freely, and for each combination, we iterate the system 1000 time steps and plot $E(1001)$ versus k_1 and k_2. Similar graphs are obtained for more than 1000 iterations. The initial values of the orbits are within 0.0001 of each other. We observe that the error is zero for smaller values of k_1 and k_2, but for larger values the error is nonzero.

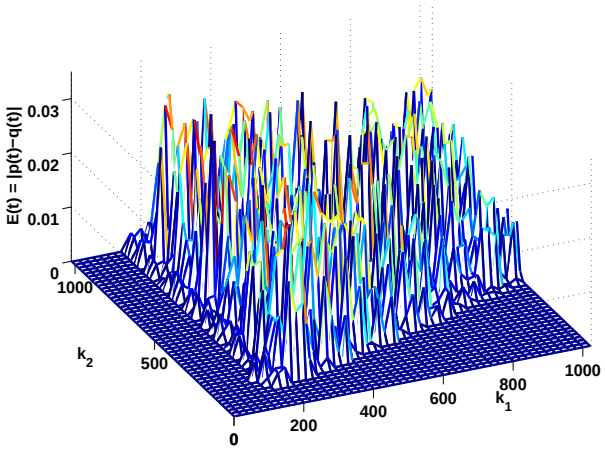


FIGURE 5. Error plot for the ARBN model with $N = 1024, M1 = 960, M2 = 64$, and x_t from a hypergeometric distribution with a total of N balls, $w = 900$ white balls, and $s = 700$ selected balls. For each combination of the parameters k_1 and k_2 we iterate the system 1000 times and plot $E(1001)$. We observe that if k_1 and k_2 are large enough the error is not zero. This situation holds for more than 1000 iterations as well.

Similarly, in Figure 6 we fix the parameters $N = 2^{10}, M_1 = 960, M_2 = 64, k_1 = 64, k_2 = 1000$, and allow the parameters w and s to move freely. Again, we iterate the system 1000 times and plot $E(1001)$ versus w and s. Again, we observe that w and s have to be large enough for the error to be nonzero. We also observe that if both w and s are very large the error is zero again. The bifurcation diagrams to follow will supplement the error plots and clarify that indeed, when the parameters are large enough the system exhibits chaos. Otherwise it has an ordered behavior.

To graph the bifurcation diagrams with integer values for the parameters k, we consider the two dimensional case and fix the parameters N, M_1, M_2. In Figure 7 we graph bifurcation diagrams for

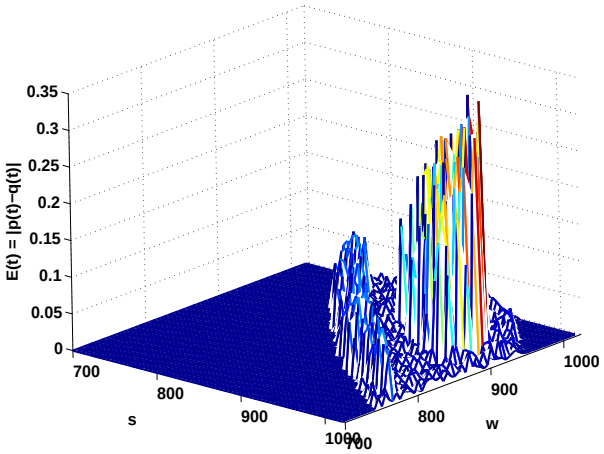


FIGURE 6. Error plot for the ARBN model with $N = 1024, M1 = 960, M2 = 64, k_1 = 64, k_2 = 1000$, and x_t from a hypergeometric distribution with a total of N balls, w white balls, and s selected balls. For each combination of the parameters w and s we iterate the system 1000 times and plot $E(1001)$. We observe that if w and s are large enough the error is not zero. This situation holds for more than 1000 iterations as well. At the same time, if both w and s are very large, the error is again zero.

$N = 1024, M_1 = 64, M_2 = 960$ and let k_1 take on a few values between 1 and 1023, while k_2 increases freely. The values x_t are from a binomial distribution with N trials and probability of success 0.5. The diagrams represent p vs. (k_1, k_2) for only a few values of k_1, which allows one to understand how the diagrams change from one value of k_1 to another. We note here that the initial values (p_1, p_2) are the same for all the "slices" shown in the graph. We observe that for smaller values of k_1 the diagrams exhibit one period doubling bifurcation which is reversed for large values of k_2. Thus the system has a very ordered behavior. Figure 8 is a zoom in on the "slice" $k_1 = 50$ of Figure 7 for more clarity. We observe that for fixed, but larger values of k_1 the situation changes and the system exhibits chaos. The route to chaos is through a cascade of period doubling bifurcations which is then reversed to a cascade of period halving bifurcations for large values of k_2. This can be seen by looking at the "slices" of Figure 7. Since the values of k_1 and k_2 are bounded above by $N-1$, if k_1 is large enough, the reversed cascade does not occur in the given range of k_2 values. However, we observe that if the number of nodes is large, the reversed bifurcations occur for larger and larger values of k_2. On the other hand, if θ is small enough, the system may exhibit only ordered behavior for any values of k_1 and k_2 within their range $\{1, 2, \ldots, N-1\}$. As θ increases, the chaos occurs starting with smaller and smaller values of k_1, and the range of values for $p(t)$ becomes wider. We include for comparison two other figures: Figure 9 is the analog of Figure 7 for $\theta = 0.4$ and Figure 10 is the analog of Figure 7 for $\theta = 0.9$.

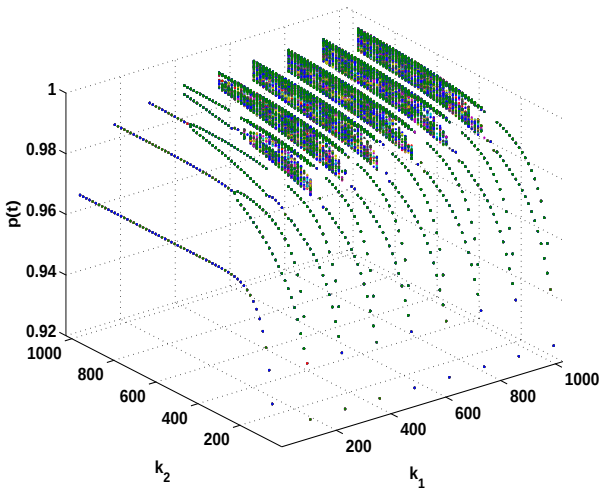


FIGURE 7. Bifurcation diagram for the case where $N = 1024, M_1 = 64, M_2, = 960$, and k_1 takes on a range of values between 1 and $N-1$ as seen in the graph, while k_2 increases freely (up to the maximum of $N-1$). The values x_t are from a binomial distribution with N trials and probability of success 0.5. We observe that for smaller values of k_1 the diagrams exhibit one period doubling bifurcation which is reversed as k_2 increases. Thus the system has a very ordered behavior. In the next figure we zoom in on the "slice" $k_1 = 50$ for more clarity. At the same time, for larger values of k_1 the system exhibits chaos as k_2 increases which occurs through period doubling bifurcations which may be reversed as k_2 becomes large. If k_1 is large enough the reversed bifurcations do not appear within the range of admissible k_2 values.

Thus for the case of a binomial distribution with N trials and probability of success θ medium or large, the system exhibits chaos for certain values of the parameters k_1 and k_2, which may be reversed through a cascade of period halving bifurcations, as seen more precisely in Figure 11. As k_1 increases the reversed bifurcations occur for larger values of k_2. At the same time, we observe that if M_1 increases, the reversed bifurcations occur for smaller values of k_2 when k_1 is fixed. Also, as the probability of success θ decreases the chaos appears for larger values of k_1 and k_2. Similar situations occur for the cases when x_t is generated by a uniform or a Poisson distribution.

For the case when x_t is from a power law distribution we observe that the diagrams "fan-out" as shown in Figures 12-13 where the parameter of the distribution is $\gamma = 4$. This is suggestive of an ordered behavior. Figure 13 is a zoom in on the slice $k_1 = 5$ of Figure 12 with various levels of detail in the three subplots provided. The diagrams are similar for any values of k_1 within the range $\{1, 2, \ldots, N-1\}$. We observe the same behavior for values of $\gamma >= 4$, while for smaller values the system exhibits only stable fixed points. These graphs are typical for various combinations of the parameters.

In the case of the hypergeometric distribution with parameters N, w, s, the bifurcation diagrams support the behavior observed in the error plots, namely, that large enough values of k_1 and k_2 combined

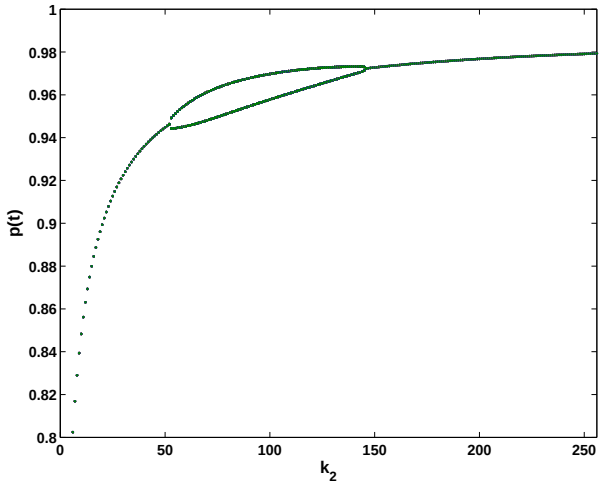


FIGURE 8. Bifurcation diagram for the case where $N = 1024, M_1 = 64, M_2, = 960, k_1 = 50$ and k_2 increases freely. The values x_t are from a binomial distribution with N trials and probability of success 0.5. We observe that the diagrams exhibit one period doubling bifurcation which is reversed for large values of k_2. Thus the system has a very ordered behavior. This figure is a zoom in on the "slice" $k_1 = 50$ of Figure 7.

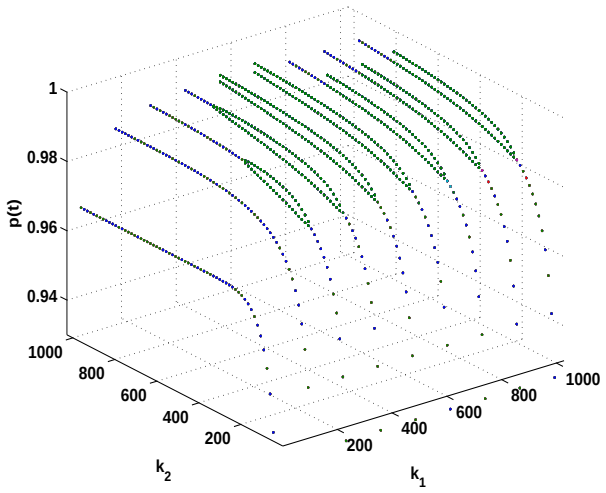


FIGURE 9. Bifurcation diagram for the case where $N = 1024, M_1 = 64, M_2, = 960$, and k_1 takes on a few values between 1 and $N - 1$ as seen in the graph, while k_2 increases freely (up to the maximum of $N - 1$). The values x_t are from a binomial distribution with N trials and probability of success 0.4. We observe that the diagrams exhibit one period doubling bifurcation which may be reversed as k_2 increases, similarly to Figure 8. Thus the system has a very ordered behavior.

with suitable values of w and s generate chaos. More precisely, if say s and w are fixed and we allow k_1 and k_2 to increase, we observe that for small values of k_1 the system exhibits order as k_2 moves freely,

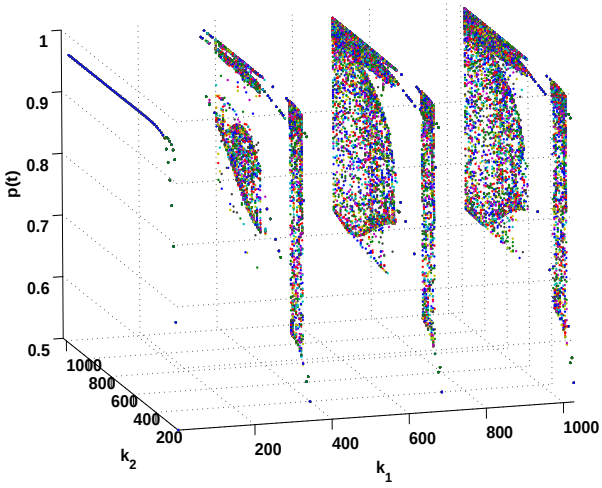


FIGURE 10. Bifurcation diagram for the case where $N = 1024, M_1 = 64, M_2 = 960$, and k_1 takes on a few values between 1 and $N-1$ as seen in the graph, while k_2 increases freely (up to the maximum of $N-1$). The values x_t are from a binomial distribution with N trials and probability of success 0.9. We observe that for a wide range of values of k_1 (between 1 and $N-1$) the system exhibits chaos as k_2 increases which occurs through period doubling bifurcations.

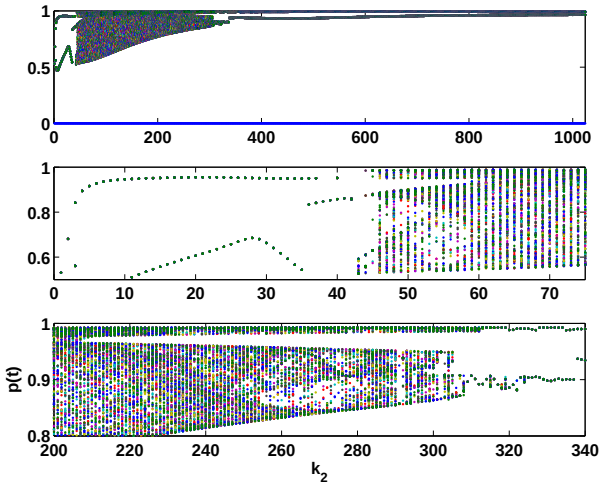


FIGURE 11. Bifurcation diagram for the case where $N = 1024, M_1 = 64, M_2 = 960, k_1 = 50$ and k_2 increases freely. The values x_t are from a binomial distribution with N trials and probability of success 0.9. We observe that the route to chaos is through a cascade of period doubling bifurcations which are reversed for larger values of k_2.

while for larger values of k_1 the system exhibits period doubling bifurcations and chaos as k_2 increases freely. This can be seen in Figure 14, where we graph bifurcation diagrams after iterating the system 2000 steps. We select only a few values of k_1 whereas k_2 increases freely. The fixed parameters are

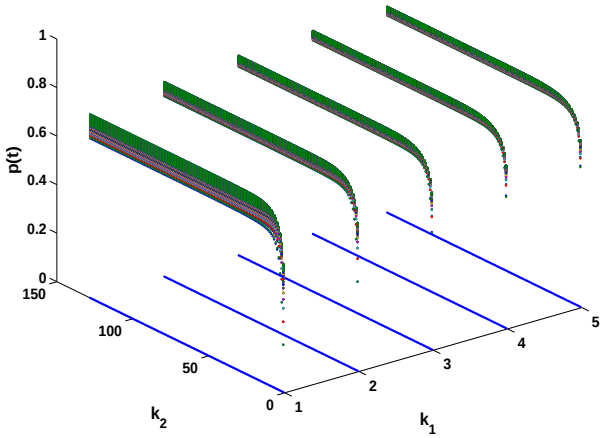


FIGURE 12. Bifurcation diagram for the case where $N = 1024, M_1 = 512, M_2 = 512, k_1 = 1, 2, 3, 4, 5$ and k_2 increases freely. The values x_t are from a power law distribution with parameter $\gamma = 4$. We observe that the diagrams show an ordered but nontrivial behavior. In the next figure we provide a zoom in on the "slice" $k_1 = 5$ to see that the diagrams are non-bifurcating.

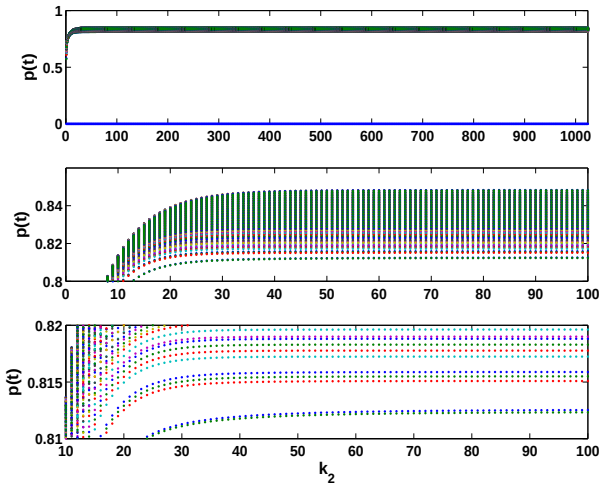


FIGURE 13. Bifurcation diagram for the case where $N = 1024, M_1 = 512, M_2 = 512, k_1 = 5$ and k_2 increases freely. The values x_t are from a power law distribution with parameter $\gamma = 4$. We observe that the diagrams show an ordered but nontrivial behavior. This figure is a zoom in on the "slice" $k_1 = 5$ of Figure 12, providing three different levels of detail.

$N = 2^{10}, M_1 = 960, M_2 = 64, w = 700, s = 900$ and they match the previous error plots for the hypergeometric distribution. At the same time, if we consider similar graphs for combinations of the parameters s and w that generate larger values of x_t, the diagrams exhibit period doubling bifurcations and chaos starting with smaller values of k_1. On the other hand, if s and w are such that the values x_t become

smaller, then the system exhibits an ordered behavior for a wider range of k_1 values, potentially for all values of k_1 in the admissible range if the values of x_t are small enough.

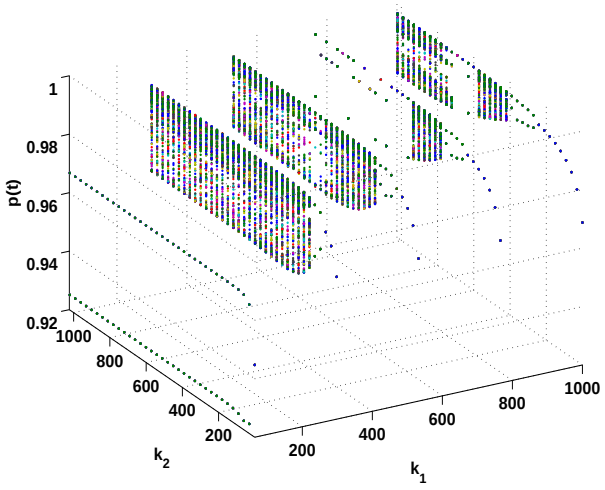


FIGURE 14. Bifurcation diagram for the case where $N = 1024, M_1 = 960, M_2 = 64$, k_1 takes on a few values in the allowed range of values, and k_2 increases freely. The values x_t are from a hypergeometric distribution with parameters $N, w = 700, s = 900$. We observe that for small values of k_1 the system exhibits order as k_2 increases, whereas when k_1 is large enough the system exhibits period doubling bifurcations and chaos.

Finally, to end the analysis we look at the J-dimensional map

$$f_j(p_1, p_2, \ldots, p_J) =$$

$$= \frac{M_j}{N}\left[\frac{p_j}{\frac{M_j}{N}}\left(1 - \frac{x_t}{N}\left(1 - \left(1 - \sum_{i=1}^{J} p_i\right)^{k_j} + \left(\sum_{i=1}^{J} p_i\right)^{k_j}\right)\right) + \frac{x_t}{N}\left(1 - \left(1 - \sum_{i=1}^{J} p_i\right)^{k_j}\right)\right]$$

where $j = 1, 2, \ldots, J$, and find its fixed points, that is we solve the system of equations $f_j(p_1, p_2, \ldots, p_J) = p_j, j = 1, 2, \ldots, J$ which yields the following equivalent system

$$p_j = \frac{\frac{M_j}{N}\left(1 - \left(1 - \sum_{i=1}^{J} p_i\right)^{k_j}\right)}{1 - \left(1 - \sum_{i=1}^{J} p_i\right)^{k_j} + \left(\sum_{i=1}^{J} p_i\right)^{k_j}}, \qquad j = 1, 2, \ldots, J. \tag{3.1}$$

It is clear that if $k_j \to \infty$ for a fixed j in the above system, then p_j converges to $\frac{M_j}{N}$. Of course the sequence $(p_j)_j$ is increasing. Also, $p_j = 0, j = 1, 2, \ldots, J$ is obviously a fixed point of the J-dimensional map. Observe that if for a given $j = 1, 2, \ldots, J$ we set $k_j = 1$ then $p_j = \frac{1}{2}\frac{M_j}{N}$. On the other hand, if none of the k's are equal to 1, since $p_j \to \frac{M_j}{N}$ as $k_j \to \infty$ the sum $\sum_{j=1}^{J} p_j \to 1$ as the k's increase. One other aspect to observe is that the fixed points are independent of the choice of x_t. In Figure 15 we observe that the fixed points converge indeed to $\frac{M_j}{N}$ for each specified combination of parameters, and to $\frac{1}{2}\frac{M_j}{N}$

for $k_j = 1$. We look at $2, 3$ and 4-dimensional cases, and we graph the values of the $p_j, j = 1, 2, \ldots, J$ and $p = \sum_{j=1}^{J} p_j$ against one of the k values which is free. The other k values are constant.

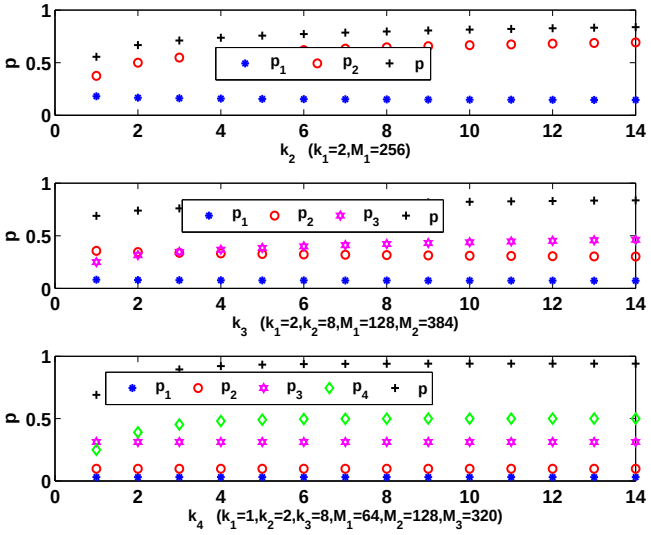


FIGURE 15. Fixed points for the following cases. *Subplot 1:* We graph $p_1, p_2, p = p_1 + p_2$ for the 2 dimensional case in which $N = 1024, M_1 = 256, M_2 = 768, k_1 = 2$. Observe that $p_1 \to \frac{M_1}{N} = 0.25$, while $p_2 \to \frac{M_2}{N} = 0.75$ as k_2 increases. *Subplot 2:* We graph $p_1, p_2, p_3, p = p_1 + p_2 + p_3$ for the 3 dimensional case in which $N = 1024, M_1 = 128, M_2 = 384, M_3 = 512, k_1 = 2, k_2 = 8$. Observe that $p_1 \to \frac{M_1}{N} = 0.125, p_2 \to \frac{M_2}{N} = 0.375$, while $p_3 \to \frac{M_3}{N} = 0.5$ as k_3 increases. *Subplot 3:* We graph $p_1, p_2, p_3, p_4, p = p_1 + p_2 + p_3 + p_4$ for the 4 dimensional case in which $N = 1024, M_1 = 64, M_2 = 128, M_3 = 320, M_4 = 512, k_1 = 1, k_2 = 2, k_3 = 8$. Observe that $p_1 = \frac{1}{2}\frac{M_1}{N} = 0.03125, p_2 \to \frac{M_2}{N} = 0.125, p_3 \to \frac{M_3}{N} = 0.3125$, while $p_4 \to \frac{M_4}{N} = 0.5$ as k_4 increases.

4. More on Bifurcations

Consider the formula (2.2) simplified by replacing the term $\frac{x_t}{N}$ by a constant $\alpha \in [0, 1]$. We can view α as the mean value of the distribution generating x_t divided by N. Thus the formula becomes

$$p_j(t+1) = \frac{M_j}{N}\left[\frac{p_j(t)}{\frac{M_j}{N}}\left(1 - \alpha\left(1 - (1 - p(t))^{k_j} + p(t)^{k_j}\right)\right) + \alpha\left(1 - (1 - p(t))^{k_j}\right)\right]$$

where $j = 1, 2, \ldots, J$. Again, let

$$p(t+1) = \sum_{j=1}^{J} p_j(t+1).$$

To understand the appearance of bifurcations in the previous section, we will consider the two dimensional case, with two possible values for the number of parents of a node, k_1 and k_2. Also, for simplicity,

we will assume $\frac{M_1}{N} = \frac{M_2}{N} = \frac{1}{2}$, and we let $p_2(t) = \frac{M_2}{N} = \frac{1}{2}$. Recall that from the fixed point condition (3.1) we have obtained that $p_2(t) \to \frac{M_2}{N}$ as $k_2 \to \infty$. Thus, assuming $k_2 \to \infty$, we obtain

$$p_1(t+1) = p_1(t)\left[1-\alpha\left(1-\left(\frac{1}{2}-p_1(t)\right)^{k_1}+\left(\frac{1}{2}+p_1(t)\right)^{k_1}\right)\right]+\frac{\alpha}{2}\left(1-\left(\frac{1}{2}-p_1(t)\right)^{k_1}\right)$$

$$p_2(t+1) = \frac{1}{2}.$$

Equivalently, using the fact that $p(t+1) = p_1(t+1) + p_2(t+1)$ we obtain

$$p(t+1) = \frac{1}{2} + \left(p(t)-\frac{1}{2}\right)\left[1-\alpha\left(1-(1-p(t))^{k_1}+p(t)^{k_1}\right)\right]+\frac{\alpha}{2}\left(1-(1-p(t))^{k_1}\right). \tag{4.1}$$

Thus we can study the behavior of this map, which corresponds to large values of k_2. We will look now at the three dimensional bifurcation diagram of the map (4.1), considering p as a function of both k_1 and α. We iterate the system 2000 before plotting the diagrams in Figures 16-17. In Figure 16 we show slices along k_1 and observe that the system is stable for small values of α, but exhibits period doubling bifurcations and chaos as α increases. We note here that similar results have been obtained in [19] for the case of generalized asynchronous random Boolean networks with a constant number of parents. Figure 16 corresponds to Figure 11 of that paper. On the other hand, in Figure 16 we observe ordered behavior for $\alpha = 1$ as well. We will elaborate on this in what follows. Figure 17 is a zoom in on the slice $\alpha = 0.8$ of Figure 16, for a wider range of k_1 values, with three levels of detail. We observe the existence of period doubling bifurcations and chaos, together with periodic windows. In fact, if we set $\alpha = 0.8$ and $k_1 = 500$, the graph of the first six iterates of the map shows the complexity introduced in the system by higher order iterates, as shown in Figure 18. For example, the third iterate shows the existence of period three orbits, and therefore, according to well-known theoretical results [26], implies the existence of periodic orbits of all periods and an uncountably infinite set of sensitive points.

We note that for $\alpha = 1$, meaning that the network is synchronous, the bifurcation diagram shows a range of values for k_1 corresponding to a single stable fixed point and one period doubling bifurcation to a single stable period two orbit. This is exactly the behavior observed in [21] for a synchronous network with varying number of parents, and corresponds to Figure 8 of that paper.

Similar to Figure 16 we can obtain bifurcation diagrams for p as a function of k and α with slices along α. This is shown in Figure 19. It is apparent that the complex behavior occurs for larger values of α, or in terms of the network, for the case when a sufficient number of nodes are updated at each time point.

We finalize the focus on the two dimensional case by looking now at the map (4.1), viewing it entirely as a map for p with one parameter k_1. We view it as a surface $p(t+1)$ versus $(p(t), k_1)$ for four different values of α in Figure 20. We note that the last of the four graphs corresponds to $\alpha = 1$, thus to a synchronous network. This last graph is similar to the one obtained in Figure 9(a) of [21].

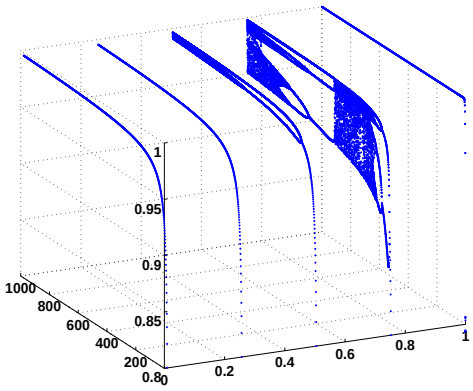


FIGURE 16. Bifurcation diagram for the map $p(t+1) = \frac{1}{2} + \left(p(t) - \frac{1}{2}\right)\left[1 - \alpha\left(1 - (1-p(t)^{k_1} + p(t)^{k_1}\right)\right] + \frac{\alpha}{2}\left(1 - (1-p(t))^{k_1}\right)$, where α takes on a few values in $[0,1]$, while k_1 increases from 1 to 1000. The vertical axis represents the 2001 iteration of $p(t)$ for various initial values. We can see the rather ordered behavior for small α as well as for $\alpha = 1$, and period doubling bifurcations and chaos for larger values of α.

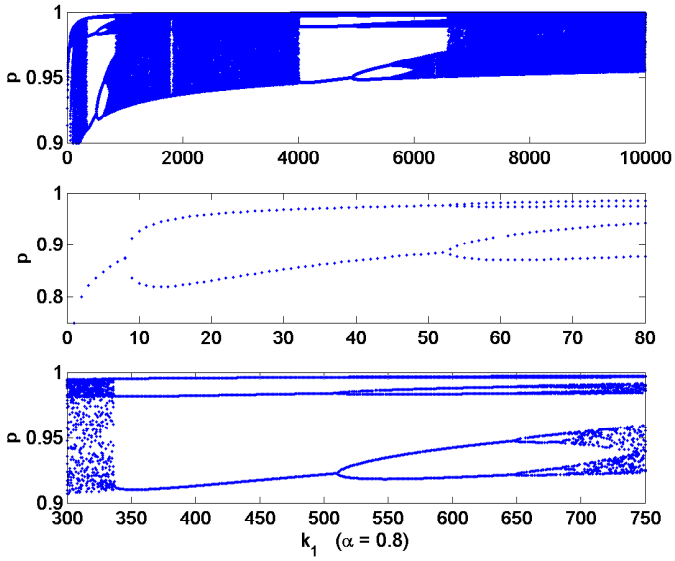


FIGURE 17. Bifurcation diagram for the map $p(t+1) = \frac{1}{2} + \left(p(t) - \frac{1}{2}\right)\left[1 - \alpha\left(1 - (1-p(t)^{k_1} + p(t)^{k_1}\right)\right] + \frac{\alpha}{2}\left(1 - (1-p(t))^{k_1}\right)$, where $\alpha = 0.8$ and k_1 increases from 1 to 10000. The vertical axis represents the 2001 iteration of $p(t)$ for various initial values. We observe the period doubling bifurcations and chaos and the periodic windows in the three levels of detail.

We can now look at higher dimension maps. For example, we can analyze the three dimensional case in a similar fashion, by considering $\frac{M_1}{N} = \frac{M_2}{N} = \frac{1}{4}$, $\frac{M_3}{N} = \frac{1}{2}$, and $p_3(t) = \frac{1}{2}$ as $k_3 \to \infty$ in the fixed point

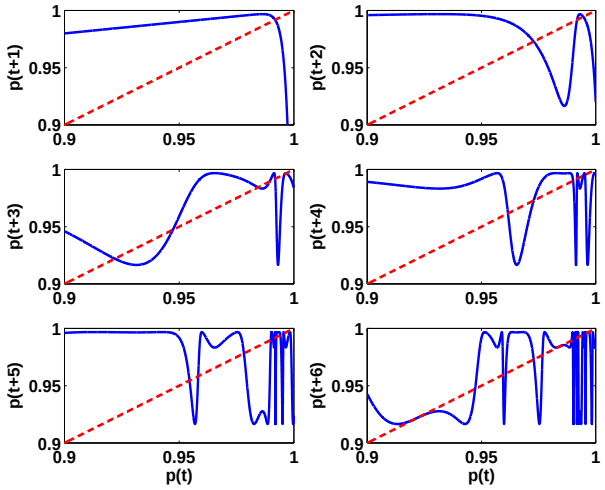


FIGURE 18. The first six iterations of the map $p(t+1) = \frac{1}{2} + \left(p(t) - \frac{1}{2}\right)\left[1 - \alpha\left(1 - (1-p(t)^{k_1} + p(t)^{k_1}\right)\right] + \frac{\alpha}{2}\left(1 - (1-p(t))^{k_1}\right)$, where $\alpha = 0.8$ and $k_1 = 500$. The higher order iterates introduce complexity in the system.

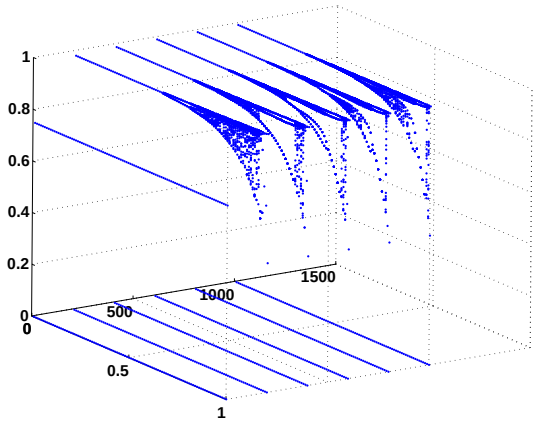


FIGURE 19. Bifurcation diagram for the map $p(t+1) = \frac{1}{2} + \left(p(t) - \frac{1}{2}\right)\left[1 - \alpha\left(1 - (1-p(t)^{k_1} + p(t)^{k_1}\right)\right] + \frac{\alpha}{2}\left(1 - (1-p(t))^{k_1}\right)$, where k_1 takes on a few values between 1 and 1000, and α moves freely in $[0,1]$. The vertical axis represents the 2001 iteration of $p(t)$ for various initial values. We can see the rather ordered behavior for small α and more complex behavior for larger values of α.

condition. The maps become

$$p_1(t+1) = p_1(t)\left[1 - \alpha\left(1 - \left(\frac{1}{2} - p_1(t) - p_2(t)\right)^{k_1} + \left(\frac{1}{2} + p_1(t) + p_2(t)\right)^{k_1}\right)\right] +$$

$$+\frac{\alpha}{4}\left(1 - \left(\frac{1}{2} - p_1(t) - p_2(t)\right)^{k_1}\right)$$

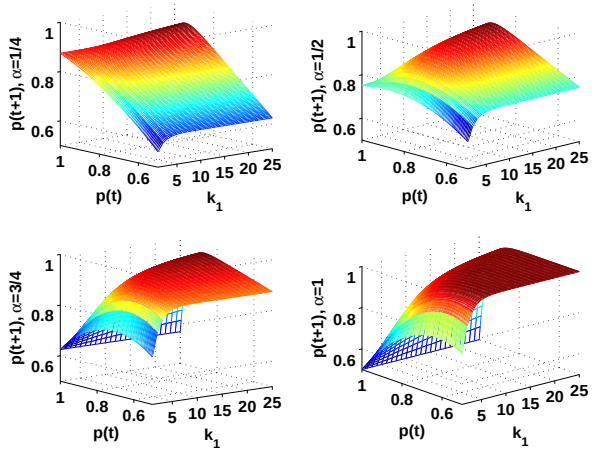


FIGURE 20. Surface map for $p(t+1)$ versus $(p(t), k_1)$ for the map $p(t+1) = \frac{1}{2} + (p(t) - \frac{1}{2})\left[1 - \alpha\left(1 - (1-p(t))^{k_1} + p(t)^{k_1}\right)\right] + \frac{\alpha}{2}\left(1 - (1-p(t))^{k_1}\right)$, viewed as a map for p with one parameter k_1. Here $\alpha = 1/4, 1/2, 3/4, 1$ respectively. We note the increased complexity of the surface as α increases.

$$p_2(t+1) = p_2(t)\left[1 - \alpha\left(1 - \left(\frac{1}{2} - p_1(t) - p_2(t)\right)^{k_2} + \left(\frac{1}{2} + p_1(t) + p_2(t)\right)^{k_2}\right)\right] +$$

$$+\frac{\alpha}{4}\left(1 - \left(\frac{1}{2} - p_1(t) - p_2(t)\right)^{k_2}\right) \tag{4.2}$$

and $p(t+1) = p_1(t+1) + p_2(t+1) + \frac{1}{2}$. Then the map $p(t)$ produces a monotone map as seen in Figure 21, where $k_1 = 5$ and $k_2 = 10$. We graph $p(t+1)$ versus $(p_1(t), p_2(t))$ for the following values of $\alpha = 1/4, 1/2, 3/4, 1$. Again we observe the similarity of the fourth graph for $\alpha = 1$ to Figure 9(b) of [21] which corresponds to the synchronous case.

Due to the analogy with the two dimensional case, we can conclude that a high connectivity value k (even for one class of nodes) can lead to a variety of behaviors, that range from order to chaos. The chaos is due to period doubling bifurcations, which in some cases may be reversed through a cascade of period halving bifurcations. For the case when a small or moderate number of nodes update at the same time, the system is ordered in the long run, exhibiting stable fixed points or period two orbits. For a large number of nodes updated at the same time, the system can exhibit order or chaos, depending on the other parameters. The chaos occurs through a cascade of period doubling bifurcations, which may be reversed into period halving bifurcations. We observe that the synchronous case is similar to the previous results obtained in [21] for the case of synchronous networks.

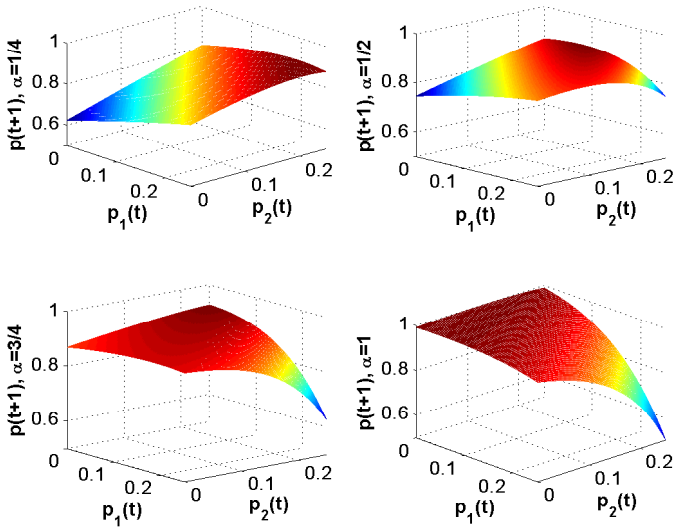


FIGURE 21. Surface map for $p(t+1)$ versus $(p_1(t), p_2(t))$ for the map system (4.2). Here $k_1 = 5, k_2 = 10$ and $\alpha = 1/4, 1/2, 3/4, 1$ respectively. We note the increased complexity of the surface as α increases.

5. Conclusions

We consider a Boolean network with N nodes, each node having a number of parents that can vary from one node to another. The Boolean rule is a generalization of ECA rule 126 and is assumed the same for all nodes. The nodes are updated asynchronously, and the number of nodes to be updated at each time point is generated by a chosen random variable. The distributions used in this paper are uniform, binomial, Poisson, power law, and hypergeometric. The network model provides the probability $p(t+1)$ of finding a node in state 1 at time $t+1$ knowing $p(t)$. Using the model we study the dynamics of the system through sensitivity of the orbits to initial values, bifurcation diagrams, and fixed point analysis. We show that the system may exhibit order or chaos, depending on the underlying parameters, the distributions used, and the number of nodes to be updated at each time point. The route to chaos is due to period-doubling bifurcations which turn into reversed bifurcations for certain combinations of parameter values. A detailed discussion on bifurcations shows that in general, when a few nodes are updated at the same time, the system exhibits order, while for the case when a large number of nodes are updated at each time point, the system could exhibit chaos which may be reversed for high values of the connectivity parameters.

Future work will extend the asynchrony to various stochastic processes such as Markov processes or (homogeneous or nonhomogeneous) Poisson processes. For a more in depth analysis of the dynamics of the model, we would consider also processes such as Brownian motion or fractional Brownian motion,

to account for short or long range dependence and self similarity properties of the number of updated nodes.

Extending this work to the entire class of "totalistic" ECA rules ([21], [23]), would be a natural step in the future, since the construction of the model in this paper depends only on the number of parent nodes that are in state 1 or 0 at each time point, and not on their spatial structure. In conjunction with this we plan on allowing for multiple Boolean rules since nodes of real systems may not behave according to a fixed rule.

It has been observed that many processes in natural or artificial networks, could be both asynchronous and ordered [27]. The authors of [27] propose the spotlight model in which the Boolean network is divided into modules, each module being associated to a regulator node which controls the updates of the module, depending on its own state. The asynchrony is obtained by altering the number of modules used. Applying the spotlight model to the network described in this paper could generate some interesting results.

Many complex systems in areas such as biology, chemistry, neural networks, or social networks, have been shown to evolve based on synchronization of coupled elements ([28] - [31]). In [32] - [34], the authors discuss the synchronization of Kauffman networks and coupled cellular automata. In light of their work, it would be of interest to study the synchronization of coupled networks similar to those described in this paper, allowing for variability in the initial states of the networks to be synchronized, as well as in the underlying parameters.

References

[1] Kauffman S.A., *Metabolic stability and epigenesis in randomly constructed genetic nets*, J. Theor. Biol., 22 (1969), p. 437-467.

[2] Glass K., Kauffman S.A., *The logical analysis if continuous, non-linear biochemical control networks*, J. Theor. Biol., 39 (1973), p. 103-129.

[3] Kauffman S.A., *The origins of order*, Oxford University Press, 1993, p. 173-235.

[4] Albert R., Barabasi A-L., *Dynamics of complex systems: scaling laws for the period of Boolean networks*, Physical Review Letters, 84, 24 (2000), p. 5660-5663.

[5] Heidel J., Maloney J., Farrow C., Rogers J.A., *Finding cycles in synchronous Boolean networks with applications to biochemical systems*, International Journal of Bifurcation and Chaos, 13 (2003), p. 535-552.

[6] Fox J.J., Hill C.C., *From topology to dynamics in biochemical networks*, Chaos, Vol. 11, 4 (2001), p. 809-815.

[7] Shmulevich I., Dougherty E.R., Zhang W., *From Boolean to Probabilistic Boolean Networks as Models of Genetic Regulatory Networks*, Proceedings of the IEEE, Vol. 90, 11 (2002), p. 1778-1792.

[8] Shmulevich I., Dougherty E.R., Kim S., Zhang W., *Probabilistic Boolean networks: a rule-based uncertainty model for gene regulatory networks*, Bioinformatics, Vol. 18, 2 (2002), p. 261-274.

[9] Shmulevich I., Dougherty E.R., Zhang W., *Control of stationary behavior in probabilistic Boolean networks by means of structural intervention*, Journal of Biological Systems, Vol. 10, 4 (2002), p. 431-445.

[10] Shmulevich I., Dougherty E.R., Zhang W., *Gene perturbation and intervention in probabilistic Boolean networks*, Bioinformatics, Vol. 18, 10 (2002), p. 1319-1331.

[11] Shmulevich I., Lähdesmäki H.Dougherty E.R., Astola J., Zhang W., *The role of certain Post classes in Boolean network models of genetic networks*, PNAS, Vol. 100, 19 (2003), p. 10734-10739.

[12] Shmulevich I., Kauffman S., *Activities and sensitivities in Boolean network models*, Physical Review Letters, Vol. 93, 4 (2004), 048701.

[13] Huang S., *Genomics, complexity and drug discovery: insights from Boolean network models of cellular regulation*, Pharmacogenomics, 2, 3 (2001), p. 203-222.

[14] Huang S., Ingber D.E., *Shape - dependent control of cell growth, differentiation, and apoptosis: switching between attractors in cell regulatory networks*, Experimental Cell Research, 261 (2000), p. 91-103.

[15] Silvescu A., Honavar V., *Temporal Boolean network models of genetic networks and their inference from gene expression time series*, Complex Systems, 13 (2001), p. 61-78.

[16] Gómez-Gardeńes J., Moreno Y., Floría L.M., *Modeling complex gene expression networks: steady, periodic and chaotic states*, Quantitative Biology, 39 (1973), p. 103-129.

[17] Kauffman S., Peterson C., Samuelsson B., Troein C., *Random Boolean network models and the yeast transcriptional network*, J. Theor. Biol., 22 (1969), p. 437-467.

[18] Andrecut M., Ali M. K., *Chaos in a simple Boolean network*, International Journal of Modern Physics B, Vol. 15, 1 (2001), p. 17-23.

[19] Matache M.T., Heidel J., *Asynchronous random Boolean network model based on elementary cellular automata rule 126*, Physical Review E 71, 2005, 026232.

[20] Gershenson C., *Classification of Random Boolean Networks*, Artificial Life VIII, Standish, Abbass, Bedau (eds) (MIT Press), 2002, p. 1-8.

[21] Matache M.T., Heidel J., *A random Boolean network model exhibiting deterministic chaos*, Phys. Rev. E 69, 056214, 2004, 10 pages.

[22] Cornforth D., Green D.G., Newth D., Kirley M., *Do artificial ants march in step? Ordered asynchronous processes and modularity in biological systems* , Artificial Life VIII, MIT Press, 2002, p. 28-32.

[23] Wolfram S., *A new kind of science*, Wolfram Media, Champaign, 2002, p. 51-114.

[24] Wuensche A., *Discrete Dynamical Networks and their Attractor Basins*, Complexity International, Vol. 6 (1999).

[25] Wuensche A., *Basins of Attraction in Network Dynamics: A Conceptual Framework for Biomolecular Networks*, Modularity in Development and Evolution, eds G.Schlosser and G.P.Wagner. Chicago University Press 2004, chapter 13, p. 288-311.

[26] Alligood K. T., Sauer T. D., Yorke J. A., *Chaos, An introduction to dynamical systems*, Springer-Verlag New York, 1997, p. 533.

[27] Cornforth, D., Green, D.G., Newth, D., Kirley, M.R., *Ordered asynchronous processes in natural and artificial systems*, P. Whigham et al. (eds). Proceedings of the 5th Australia-Japan Joint Workshop on Intelligent & Evolutionary Systems, The University of Otago, Dunedin, New Zealand, 2001, p. 105-112.

[28] Winfree A.T., *The geometry of biological time*, Springer, Berlin, 1980.

[29] Khrustova N., Veser G., Mikhailov A., Imbihl R., *Delay-induced chaos in catalytic surface reactions: no reduction on Pt(100)* , Phys. Rev. Lett. 75, 1995, p. 3564-3567.

[30] Abarbanel H.D., Rabinovich M.I., Selverston A., Bazhenov M.V., Huerta R., Sushchik M.M., Rubchinskii L.L., *Synchronization in neural networks*, Phys. Usp. 39, 1996, p. 337-362.

[31] Neda Z., Ravasz E., Vicsek T., Brechet Y., Barabasi A.L., *Physics of the rhythmic applause*, Phys. Rev. E 61, 2000, p. 6987-6992.

[32] Morelli L.G., Zanette D.H., *Synchronization of stochastically coupled cellular automata*, Phys. Rev. E 58 (1), 1998, R8.

[33] Morelli L.G., Zanette D.H., *Synchronization of Kauffman networks*, Phys. Rev. E 63, 2001, 036204-1.

[34] Zanette D.H., Morelli L.G., *Synchronization of coupled extended dynamical systems: a short review*, International Journal of Bifurcation and Chaos 13 (4), 2003, p. 1-16.