Dynamics of Random Boolean Networks with Constant Number of Parents and Asynchrony Generated by Stochastic Processes

Huimin Geng
Advisor: Dora Matache

Department of Computer Science
University of Nebraska at Omaha
Omaha, NE 68182, USA
hgeng@mail.unomaha.edu

Abstract This paper considers a simple Boolean network with \( N \) nodes, each node’s state at time \( t \) being determined by a certain number \( k \) of parent nodes, which is fixed for all nodes. The nodes, with randomly assigned neighborhoods, are asynchronously updated based on various stochastic processes. We make use of a Boolean rule that is a generalization of rule 126 of elementary cellular automata. We provide the formula for the probability of finding a node in state 1 at a time \( t \) for the class of Generalized Asynchronous Random Boolean Networks (GARBNs) in which a random number of nodes can be updated at each time point. Poisson and random walk processes are used as the random number generators. We use simulation methods to generate consecutive states of the network for both the real system and the models under the various parameter combinations and processes. The results match well. We study the dynamics of the models through sensitivity of the orbits to initial values, bifurcation diagrams, and fixed point analysis. We show that the GARBNs have behaviors that range from order to chaos depending on the type and the parameter combinations of the stochastic processes used to determine the number of nodes to be updated. We observe that for some parameter combinations, the system becomes eventually synchronous. We explain this behavior and the associated study of the dynamics.

1. Introduction

Boolean networks are networks of nodes that can be in one of two possible states ON or OFF (1 or 0), and whose evolution from one time point to another is governed by given Boolean rules.
Each node’s evolution is influenced by the state of other nodes called its parents. If all the nodes are updated at the same time then the network is synchronous, otherwise it is asynchronous.

In recent years, Boolean networks have been used extensively as models for complex networks such as genetic or biochemical networks, networks in artificial life, biophysics, condensed matter and solid-state physics, or statistical mechanics ([1], [2], [3], [4], [5], [6], [7], [8]). These models, besides being easy to understand, are relatively easy to handle. Originally introduced by Kauffman ([9], [10], [11]). The Boolean network models appeal to any situation in which the activity of the nodes of the network can be quantized to only two states, ON and OFF, and each node updates its state based on logical relationships with other nodes of the network. For example, in biological systems, genes are not independent, but rather regulate each other and act collectively (this collective behavior can be observed using microarrays). The interrelationships among genes constitute gene regulatory networks. For practical approximation, gene regulatory networks have been treated with a Boolean formalism where genes are often described as being on and off, upregulated and downregulated, etc. ([4], [5], [7], [8], [12], [13], [14], [15], [16], [17], [18]). Although a Boolean network model may represent a very simplified view of a network, it retains in most cases meaningful information that can be used to study the dynamics of the network and make inferences regarding the real system they model. Knowing the long-run behavior of such networks would allow one to identify steady-state behavior associated with tumors, and develop a methodology for altering this steady-state as means of therapy.

In this paper we analyze a random Boolean network whose dynamics are established by a generalization of ECA Rule 126 ([20]). Rule 126 is most simply described as

```
■■■ ■■□ ■□■ ■□□ □■■ □■□ □□■ □□□
↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓
□ ■ ■ ■ ■ ■ ■ □
```

where black is ON and white is OFF. Rule 126 is useful as a conceptual model of (biological) cell growth and of a (chemical) catalytic process because the central site survives (or is born) unless the neighborhood is too poorly populated or too crowded, in which case it dies. Other ECA rules such as 22, 90, and 150 have similar interpretations ([21], [22], [23]). It is interesting that Rule 126 is both a very simple growth model and yet exhibits a quite sophisticated dynamic behavior.

Recent research has focused on finding formulas for the probability of a node being ON at time \(t\) in certain random Boolean networks, and using these formulas to study the dynamics of
the networks. In [1], Andrecut and Ali have studied the complex dynamics of a simple Boolean network with \( N \) nodes, each node being influenced by exactly \( k \) other nodes. The network is updated synchronously and the Boolean rule for each node is determined only by the state of the \( k \) parents. They showed the existence of chaos in their model and that the route to chaos is through period-doubling bifurcations. In [21], Matache and Heidel present a significant generalization of the work by Andrecut and Ali. They used the same Boolean rule as Andrecut and Ali, but extended the model to the case of non-constant number of parents. They used simulation methods to generate consecutive states of the network for both the real system and the model and studied the dynamics of the model through sensitivity of the orbits to initial values, bifurcation diagrams, and fixed point analysis. They concluded that the route to chaos is due to a cascade of period-doubling bifurcation which turn into reversed (period-having) bifurcation for certain combinations of parameter values.

One important aspect of both the studies mentioned above is that the networks are assumed to be synchronous, that is the nodes update their states at the same time. However, various authors have observed that for many biological phenomena or cellular automata, asynchronous versions are more plausible models. For example, asynchronous activity of the neurons in the brain could lead to some global patterns [24]. In [25], Matache and Heidel extended the previous work on synchronous random Boolean networks governed by a generalization of ECA Rule 126 ([1], [21]) to the case of asynchronous updating. There are various types of updating schemes in the literature ([26], [27], [28], [29], [29]). It has been shown [24] that properties of the models are changed by the particular update scheme chosen.

At the same time the random Boolean networks have been classified by Gershenson [30]. According to this author the class of Asynchronous Random Boolean Networks (ARBNs) incorporates all the cases in which at each time point a single node is selected, either at random or according to a deterministic updating rule, in order to be updated. He then generalizes the class of ARBNs to the Generalized Asynchronous Random Boolean Networks (GARBNs) defined as ARBNs which can update any number of nodes, picked at random, at each time step.

Matache and Heidel have studied the dynamics of ARBNs and GARBNs under the assumption of a fixed number of parents, a generalized ECA rule 126, and asynchrony generated by certain random variables [21]. They showed that the ARBNs generate an ordered behavior regardless of the updating scheme 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. For binomial distributions with large probability of success, the system can exhibit order or chaos for certain values of the parameter \( k \). For binomial distributions with small and medium probability of success, negative binomial, uniform distribution, or power law distributions, the behavior is ordered.

In this paper, we go one step further with the previously mentioned study, and analyze the behavior of GARBNs using various stochastic processes as random number generators for the number \( x_t \) of nodes to be updated at each time point \( t \). We focus on the Poisson process and the random walk process [31].

Poisson processes are one of the most important classes of stochastic processes, and find applications in diverse areas of science. To explain a Poisson process, we first introduce a counter which counts the number of occurrences from a starting point, and set \( X(t) = \text{number of occurrences in the interval } (0, t] \). The stochastic process \( X(t), t \geq 0 \) is called a counting process. The counting process is said to be a Poisson process having rate \( \lambda (\lambda > 0) \), if (i) \( X(0) = 0 \) (ii) the process has independent increments (iii) the number of events in any interval of length \( t \) is Poisson distributed with mean \( \lambda t \). For Poisson processes, the number of occurrences by time \( t \) follows the probability distribution function

\[
p_X(t)(x) = P(X(t) = x) = e^{-\lambda t} \frac{\lambda t^x}{x!}, \quad x = 0, 1, 2, \ldots
\]

where \( \lambda \) is the intensity of the occurrences. Figure 8 in section 3 gives a sample path of a Poisson process with \( \lambda = 0.5 \).

A random walk process is a special case of Markov chain whose state space consists of the integers \( i = 0, \pm 1, \pm 2, \ldots \), and with transition probabilities given by

\[
p_{i,i+1} = p = 1 - p_{i,i-1}, \quad i = 0, \pm 1, \pm 2, \ldots
\]

where \( 0 < p < 1 \). In another words, we may think of it as being a model for an individual walking on a straight line who at each point of time either takes one step to the right with probability \( p_{i,i+1} = p \) or one step to the left with probability \( p_{i,i-1} = 1 - p \). We consider the parameters of the random walk process as follows: (i) starting point, which could be anywhere between 0 and \( N \); (ii) the number of walking steps (the walk could be symmetric, that is the same number of the steps towards the two directions, or asymmetric, that is a different number of the steps towards left and right; (iii) the probability of walking to the right (\( p \)) and to the left (\( 1 - p \)). If the walk is out of the range \([0, N]\), we reset the values to 0 or \( N \), respectively.
Figure 15 in section 4 gives a sample path of a random walk process with the starting point at 64, symmetric walking of stride one with probability 0.5 to the left or to the right.

In section 2, we describe the network model. The formula for the probability of finding a node in state 1 is provided and depends on the number of nodes updated at each time point, and hence depends on which stochastic process is used as the number generator, and the parameter combinations. Section 3 is dedicated to a discussion of the simulation of the real system and the model, the sensitivity of the orbits to initial values and bifurcation diagrams, when the Poisson process is used as the number generator. In section 4, we continue the discussion on the above analysis but in the case of the random walk process. The fixed point analysis for the model is given in Section 5. Section 6 provides conclusions and potential further directions.

2. The Network Model

Consider a network with $N$ nodes. Each node $c_n$, $n = 0, 1, 2, \ldots, N - 1$ can take on only two values 1 or 0. Often this is interpreted as a system in which each node can be either ON or OFF. At each time point $t$ the system can be in one of the $2^N$ possible states. If all the nodes update their value at the same time the network is synchronous, otherwise it is asynchronous. The evolution of the nodes from time $t$ to time $t + 1$ is given by a Boolean rule which is considered the same for all nodes. Each node $c_n$ is assigned a random "neighborhood" of parents, whose values at time $t$ influence the value of $c_n$ at time $t + 1$ through the following Boolean rule. If $c_n$ 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 (\cite{20}, \cite{21}).

The parents of a node are chosen randomly from the remaining $N - 1$ nodes and do not change thereafter. More precisely, if a node has $k$ parents, then a set of $k$ nodes is chosen from the remaining $N - 1$ nodes with probability $\frac{1}{k} \binom{N - 1}{k}$.

Observe that the quantity $N_1(t) := \sum_{n=0}^{N-1} c_n(t)$ gives the number of nodes that are in state 1 at time $t$. The concentration of nodes in state 1 is given by $\frac{1}{N} \sum_{n=0}^{N-1} c_n(t)$. We are interested in looking at the probability $p(t + 1)$ that a node is in state 1 at time $t + 1$. In \cite{25}, the authors give Boolean network models for both Asynchronous Random Boolean Networks (ARBNs), in which only one node is updated, and Generalized Asynchronous Random Boolean Networks (GARBNs), in which any number of nodes, picked at random, can be updated at each time point. For GARBNs, $p(t + 1)$ depends on $x_t$, the number of nodes to be updated at time $t$. $x_t$ is generated according to a given discrete random variable $X$ with values $1, 2, \ldots, N$. In this
paper, we generate $x_t$ using the Poisson process and the random walk processes. Then the $x_t$ nodes are selected randomly (that is any collection of $x_t$ nodes has the same probability of being chosen). In [25] it is shown that the probability of finding a node in state 1 at time $t + 1$ is

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

In this paper we study the behavior of the system of GARBNs using the Poisson process and random walk process as random number generators for the number $x_t$ of nodes to be updated at each time point $t$. Our conclusions are the result of many simulations for various combinations of the parameter values.

3. GARBN’s with Asynchrony Generated by a Poisson Process

First we look at the case of the Poisson process as the random number generator. $x_t$, the number of nodes updated at each time point, depends on both $\lambda$ and $t$. Since $\lambda_t$ increases with time, eventually $x_t = N$ and the system will reach synchrony. When $x_t$ exceeds the number of nodes $N$ as the time increases, we set it equal to $N$. So, as time passes, the behavior of the system goes from asynchrony to synchrony. We pay more attention on the asynchronous behavior in this paper and show how the changes occur when the asynchronous system becomes synchronous.

It is useful to provide some simulations to see how well the model matches the real system. The simulations that follow in this paper have been obtained by running Matlab programs. Although we present only a few graphs in this paper, the conclusions have been drawn from numerous simulations. In general we present only typical graphs. The graphs in Figure 1 represent simulations of the model and the actual Boolean system for the case of $N = 256$ with $k = 128$ and $\lambda = 0.05$. There are 9 different graphs representing iterations of the system and the model, namely we graph $p(t + \text{iteration})$ versus $p(t)$ for $\text{iteration} = 2^i, i = 0, 1, 2, \ldots, 8$. We can deduce the behavior of the system and the model for other cases from these graphs, since all the other simulations obtained by the authors for various parameter combinations are quite similar to those in Figure 1.

We make the following observations. There is an excellent match between the model and the system for iterations $2^0 - 2^8$. For a higher number of iterations the match is also good, and both the system and the model settle around a certain value of $p(t)$ suggesting that no matter what the initial conditions are, in the long run there is either an absorbing state or cycles of states that differ only slightly in terms of the number of nodes that are 1. That is, in the long run, the
Figure 1. Iterations of the system and the model for GARBN with the Poisson process, where $N = 256$, $k = 128$ and $\lambda = 0.05$. We plot some of the first 256 iterations of the system and the model as specified in the labels. We observe the perfect match of the system and the model. After approximately 256 iterations, the model and the system reach a steady behavior with $p(t) \simeq 0.9725$.

The probability of finding a node in state 1 is approximately the same regardless of the initial state of the system.

This behavior is observed to be independent of the number of nodes or parents of a node or the value of the parameter $\lambda$ (only for small and medium values, we discuss the behavior for large values of $\lambda$ later), however, different setting of the parameters affects the fixed value of the probability $p(t)$ and the rate of convergence towards this probability. The comparisons of Figure 1 with Figure 2 and Figure 3 show this effect. In Figure 2, $N$ and $k$ are fixed (having the same value as in Figure 1: $N = 256, k = 128$), while $\lambda$ is a varied ($\lambda = 0.05$ in Figure 1 instead of $\lambda = 0.2$ in Figure 2). We find that the model and the system reach a steady behavior after 128 iterations with $\lambda = 0.2$ in Figure 2, but after 256 iterations with $\lambda = 0.05$ in Figure 1. The probability $p(t)$ in both conditions seems to converge to the same value, $p(t) \simeq 0.9725$. We have tested many different $\lambda$ values and the following tendency is observed: the bigger the $\lambda$, the faster the convergence. Figure 3 demonstrates the effect when the number of nodes $N$ and the parameter $\lambda$ are fixed (having the same value as in Figure 1: $N = 256, \lambda = 0.05$) but the number of parents $k$ changes ($k = 4$ in Figure 1 and $k = 128$ in Figure 3). The smaller the $k$, the smaller the value to which $p(t)$ converges in the long run. Similarly, if $k$ and $\lambda$ are the same, the larger the $N$, the smaller the long run value of $p(t)$. 
Figure 2. Iterations of the system and the model for GARBN with the Poisson process, where $N = 256$, $k = 128$ and $\lambda = 0.2$. We plot some of the first 256 iterations of the system and the model as specified in the labels. We observe the perfect match of the system and the model. After approximately 128 iterations, the model and the system reach a steady behavior with $p(t) \simeq 0.9725$. The bigger the $\lambda$, the faster the rate of convergence.

Figure 3. Iterations of the system and the model for GARBN with the Poisson process, where $N = 256$, $k = 4$ and $\lambda = 0.05$. We plot some of the first 256 iterations of the system and the model as specified in the labels. We observe the perfect match of the system and the model. After approximately 256 iterations, the model and the system reach a steady behavior with $p(t) \simeq 0.755$. The smaller the $k$, the smaller the steady value $p(t)$. 
Given the good match between the model and the real system, we start the study of the dynamics of the system with the sensitivity of the orbits to the initial values. We fix the parameters $N$, $k$ and $\lambda$, and choose two initial values $p(0)$ and $q(0)$ as starting points for the orbits. We iterate many times the equation of the model and compute $p(t)$ and $q(t)$ for each time point $t$. Then we plot the error $E(t) = |p(t) - q(t)|$ versus $t$. Figure 4 shows the case of $N = 4096$, $k = 512$ and $\lambda = 0.5$. This graph is typical and very similar for any other combinations of parameters considered in the experiments, including small or large values for $N$ and $k$ and small or medium values for $\lambda$. We observe that the error converges to zero at a faster rate for smaller values of $k$ and a slower rate for larger values of $k$ for fixed $N$ and $\lambda$. Also, as $N$ increases the behavior is the same, but in general the convergence rates are slower. When $N$ and $k$ are fixed, the error converges to zero faster as $\lambda$ increases. For the three graphs in the Figure 4, $p(0) - q(0) = 0.5, 0.01,$ and $0.0001$ respectively. We see that it does not matter how far apart the initial values are, since the error will eventually converge to zero.

In order to clarify even more the situation suggested by the sensitivity of the orbits to the initial values, we construct bifurcation diagrams with integer values for the parameter $k$. We fix the number of nodes $N = 512$ and $\lambda = 0.05$ in Figure 5, and we iterate the function $f(p)$ a number of times for various initial values of $p$ and plot the bifurcation diagrams after $N/8, N/4, N/2$, and $N$ iterations, respectively. We observe that there is a transient period for a reduced number of iterations, but after significant iterations the bifurcation map converges to a
value that gets closer and closer to 1 as $N$ and $k$ increase for a fixed $\lambda$. Thus in the long time, the system exhibits a very ordered behavior. Another observation is that the system reaches the stable point faster for smaller values of $\lambda$ and slower for larger values of $\lambda$.

**Figure 5.** Bifurcation diagram for the GARBN model of the Poisson process, with $N = 512$ and $\lambda = 0.05$. The model is iterated a number of times, as specified in each graph, before plotting the values of $p(t)$, to understand how the transient phase behaves. We observe the ordered behavior of the system.

When $\lambda$ is large (close to or greater than 1) in the Poisson process, the system behaves differently. For a fixed time point $t$, the number of nodes updated, $x_t$, is from a Poisson distribution with $\lambda_t = \lambda * t$. If $\lambda$ is big ($\lambda = 1$, for example), and $t$ is big ($t = N$, for example), the Poisson distribution has the parameter $\lambda_t = N$. That is, the mean of the number of nodes updated at that time is $N$. All nodes are updated so the asynchronous Boolean network becomes synchronous.

We provide in Figure 6 the simulations of the real system and the model in which the values $x_t$ are generated according to the Poisson process with $N = 256$, $k = 4$, and $\lambda = 0.9$. When the number of iterations is small or medium ($2^0 \sim 2^7$), the real system and the model match well and they reach a fixed value of $p(t)$. After approximately $N$ iterations, the system doesn’t converge to a fixed value, but rather oscillates in a certain range of values of $p(t)$. We note that since $\lambda = 0.9$ is quite large, the values $x_t$ generated with the Poisson process are mainly equal to $N$. This makes the system synchronous and therefore it behaves like a synchronous network in the long run. The results obtained in [1] will be valid in this case. We note that the model in this paper coincides with the model in [1] when $x_t = N$. 
The bifurcation diagram, as shown in Figure 7 ($N = 256$ and $\lambda = 0.9$), shows the behavior changes from the asynchrony to the synchrony. The system has an ordered behavior for $t < N/2$, reaches a chaotic behavior after about $N$ iterations, and finally ends with an ordered behavior when $t > 2N$. Figure 8 demonstrates the sample path of $x_t$ (used in the graphs of Figure 7) generated by a Poisson process with $\lambda = 0.9$ and $N = 512$.

In conclusion, when $x_t$ is generated by a Poisson process with parameter $\lambda$, the system exhibits order when $\lambda$ is small. For large value of $\lambda$ the system may exhibit ordered or chaotic behavior depending on the amount of time the system runs. Since the system becomes eventually synchronous, the long run behavior is best described by a synchronous system. The authors of [1] have shown that the system exhibits chaos in this case (they have only studied the behavior for $k < 32$), which is confirmed by our study. In addition, our study shows that the system exhibits order when the connectivity $k$ is large.

We note that even if $\lambda$ is very small, the system may eventually reach synchrony, potentially after a very long time. It would be of interest to look at situations when synchrony is not necessarily reached in any finite time. For example, allowing $\lambda$ to vary would be a natural option, thus involving non-homogeneous Poisson processes. This will make the object of future investigation.
Figure 7. Bifurcation diagram for the GARBN model of the Poisson process with $N = 512$, and $\lambda = 0.9$. The model is iterated many times to understand how phases switch between chaos and order. We observe the ordered behavior around $N/2$ iterations and chaotic behavior around $N$ iterations, and finally the model behaves the same as the synchronous one by reaching a stable $p(t)$ after $2N$ iterations.

Figure 8. A Poisson sample path of $x_t$ with $N = 512$ and $\lambda = 0.9$. The mean $\lambda_t$ and the number of nodes updated at each time point, $x_t$, increase with time $t$. $x_t$ finally stays at $N = 512$ and the system becomes synchronous.

4. GARBN’s with Asynchrony generated by a Random Walk Process

Next, we study the system behavior in the case when $x_t$ is given by a random walk process. Simulations of iterations of the system and the model show that the model is a good approximation for the Boolean system. Figure 9 gives an example of the simulation with $N = 256$, $k = 16$, 


and $x_t$ is from a random walk process with starting point $N/2$ and symmetric steps with equal probability.

The study of the sensitivity of the orbits to initial values in the random walk process shows that the error converges to zero at a faster rate than for the Poisson process (see Figures 4 and 10 for comparison). With the fixed starting point and walking steps, it is observed that
as the number of nodes $N$ increases the rate of convergence decreases, and as the number of parents $k$ increases for a fixed $N$, the rate of convergence increases too. Intuitively this makes sense since the more nodes are in the network the more time seems to be needed to reach a steady state. At the same time, for a fixed $N$, if the nodes have more parents, then the node interaction is more elaborate and speeds up the process of reaching a steady behavior. The bifurcation diagrams (Figure 11) obtained by the random walk process have similar patterns as the Poisson process, except that less iterations are required to reach the stable state in the random walk process.

We note that when the starting point is closer to $N$ or the probability to the right is bigger so that the walk is lead towards $N$ faster, the error converges to zero much slower (Figure 12) as compared with Figure 10. The bifurcation diagram in Figure 13 indicates that the system has a chaotic behavior if the random process’ parameters are such that synchrony is reached. We observe that synchrony is reached after about 150 iterations. The last two graphs in Figure 13 correspond to the synchronous case and they match the behavior observed in [1].
Figure 11. Bifurcation diagram for the GARBN model in the case of the random walk process with \(N = 512\), starting point at \(N/4\), and symmetric steps with probabilities 0.8 to the left and 0.2 to the right. We observe the ordered behavior of the system.

Figure 12. Error plot for the GARBN model in the case of the random walk process with \(N = 512\), \(k = 150\), starting point at \(N/2\), and symmetric steps with probabilities 0.2 to the left and 0.8 to the right. In each graph we plot the error \(E(t) = |p(t) - q(t)|\) versus \(t\). We start with initial values \(p(0)\) and \(q(0)\) that are 0.5, 0.01, and 0.0001 apart respectively. The error settles to zero after many iterations.

In Figure 14 we observe that the system has a very ordered behavior when the random walk parameters are such that the values \(x_t\) are not very large. The system is asynchronous and the
Figure 13. Bifurcation diagram for the GARBN model in the case of the random walk process with $N = 128$, starting point at $N/2$, and symmetric steps with probabilities 0.3 to the left and 0.7 to the right. The model is iterated a number of times to understand how phases change. We observe the chaotic behavior of the system. Note that synchrony is reached after about 150 iterations. The last two subgraphs correspond to the synchronous case and they match the behavior observed in [1].

Random walk starts at $N/2$ with probability of 0.5 of going to the left or to the right one step. Figure 15 is the sample path of $x_t$ used in the graphs in Figure 14 for symmetric random walk with probability 0.5 of going one step to the left or to the right.

Figure 14. Bifurcation diagram for the GARBN model in the case of the random walk process with $N = 128$, starting point at $N/2$, and symmetric steps with probabilities 0.5 to the left and to the right. We observe the ordered behavior.
In conclusion, when $x_t$ is generated by a random walk process, if the parameters are such that $x_t$ is not very large and the system doesn’t reach synchrony, the system has an ordered behavior, with one stable fixed point or period two orbits. On the other hand, if the parameters are such that the $x_t$ values are large and the system reaches synchrony eventually, the long run behavior suggests chaos through period doubling bifurcations for smaller values of the parameter $k$, and ordered behavior for large connectivity $k$.

5. Fixed Points Analysis of the Asynchronous Boolean Network Model

Finally, to end the analysis of the above asynchronous updating schemes we look at the map

$$f(p) = p + \frac{x_t}{N} (1 - p - p^{k+1} - (1 - p)^{k+1})$$

and find its fixed points, that is we solve the equation $f(p) = p$. This leads to the equation in $p$

$$1 - p - p^{k+1} - (1 - p)^{k+1} = 0.$$  

It is clear that if $k \to \infty$ in the above equation, we obtain $p = 1$. Also, $p = 0$ is obviously a fixed point of the map. Figure 16 shows this fact.

6. Conclusions

In this paper we consider a Boolean network with $N$ nodes, each node having $k$ parents. We use a unique Boolean rule for all the nodes, which generalizes rule 126 of cellular automata. We
study the behavior of the Generalized Asynchronous Random Boolean Networks (GARBNs) in which a random number of nodes can be updated at each time point. Stochastic processes, such as Poisson processes and random walk processes, are used as the random number generator for the number of nodes to be updated at each time point. The simulation of the model and the real system match well under different parameter combinations. We use the model to describe the system behavior through error plots, bifurcation diagrams and fixed point analysis. We show that the GARBNs scheme generates mainly order but possibly chaos depending on the parameters of the processes.

One possibility for future work consists on enlarging the class of stochastic processes as the random number generators for the number of nodes to be updated at each time point, such as a non-homogeneous Poisson process and a birth and death process. It would be of interest to extend the study to the models in which processes such as Brownian motion and even Fractional Brownian motion are involved, to account for possible long-term correlations. Also, considering various Boolean rules for the nodes is of interest, since nodes of real systems usually do not behave according to a fixed rule.

**References**


