Complexity

Boolean Networks and Their Applications in Science and Engineering

Lead Guest Editor: Jose C. Valverde
Guest Editors: Henning S. Mortveit, Carlos Gershenson, and Yongtang Shi

Boolean Networks and Their Applications in Science and Engineering

Complexity

Boolean Networks and Their Applications in Science and Engineering

Lead Guest Editor: Jose C. Valverde
Guest Editors: Henning S. Mortveit, Carlos Gershenson, and Yongtang Shi

Contents

Hindawi
Complexity
Volume 2020, Article ID 6183798, 3 pages
https://doi.org/10.1155/2020/6183798

Editorial

Boolean Networks and Their Applications in Science and Engineering

Jose C. Valverde,[1] **Henning S. Mortveit**,[2] **Carlos Gershenson**,[3,4] **and Yongtang Shi**[5]

[1] *University of Castilla-La Mancha, Albacete, Spain*
[2] *University of Virginia, Virginia, USA*
[3] *Universidad Nacional Autonoma de Mexico, Mexico City, Mexico*
[4] *ITMO University, St. Petersburg, Russia*
[5] *Center for Combinatorics and LPMC, Nankai University, Tianjin, China*

Correspondence should be addressed to Jose C. Valverde; jose.valverde@uclm.es

Received 26 June 2019; Accepted 26 June 2019; Published 8 January 2020

In recent decades, Boolean networks (BN) have emerged as an effective mathematical tool to model not only computational processes, but also several phenomena in science and engineering. For this reason, the development of the theory of such models has become a compelling need that has attracted the interest of many research groups in recent years. Dynamics of BN are traditionally associated with complexity, since they are composed of many elemental units whose behavior is relatively simple in comparison with the behavior of the entire system.

BN are a generalization of other relevant mathematical models, which appeared previously as cellular automata (CA), inspired by von Neumann and studied by Wolfram and others to explore the computational universe, or Kauffman networks (KN), proposed by Kauffman in 1969 for modeling gene regulatory networks. This gives an idea of the versatility of this new paradigm in applications to several branches of science (mathematics, physics chemistry, biology, ecology, etc.) and engineering (computing, artificial intelligence, electronics, circuits, etc.).

The aim of this special issue was to collect cutting-edge research on the different models of BN (deterministic and nondeterministic, synchronous and asynchronous, homogenous and non-homogenous, directed and undirected, regular and non-regular, etc.). Thus, several research groups in this field submitted their recent developments and future research directions concerning new models. In addition, original research articles showing some applications of BN in science and engineering were received.

Although fifteen manuscripts were submitted to the special issue, only nine of them were finally accepted for publication after the review process. These contributions are briefly described below.

In the paper "Predecessors Existence Problems and Gardens of Eden in Sequential Dynamical Systems", Aledo et al. deal with network models which are deterministic, asynchronously updated, homogeneous and defined over arbitrary (non-regular) undirected graphs, so extending the work on the synchronous case [1]. In particular, the local functions are restrictions of a global operator, given by a maxterm or minterm Boolean function; and the update order is a vertex indexed permutation. For these kinds of models, which are usually known as sequential dynamical systems (SDS) on maxterm and minterm Boolean functions, the authors solve the predecessor-existence problems algebraically. In addition, they give a characterization of the Garden-of-Eden configurations and provide the best upper bound for the number of such configurations.

The article "A Boolean network approach to estrogen transcriptional regulation" by Anda-Jáuregui et al. presents a dynamical model of gene regulation of the Estrogen receptor transcription network based on known regulatory interactions, to better understand the implications of deregulation of the Estrogen and Estrogen receptor regulatory networks. By using an adaptation to classical Boolean Networks dynamics,

the authors identify proliferative and anti-proliferative gene expression states of the network. They also identify key players that promote these altered states when perturbed. In addition, they model how pairwise gene alterations may contribute to shifts between these two proliferative states. Furthermore, they find that the coordinated subexpression of E2F1 and SMAD4 is the most important combination in terms of promoting proliferative states in the network.

The paper "Binomial representation of cryptographic binary sequences and its relation to cellular automata" by Cardel and Fuster-Sabater is devoted to studying some properties of binomial sequences. In particular, they show how any binary sequence whose terms are repeated periodically with period a power of two can be decomposed by some binomial sequences. Furthermore, the authors analyze other interesting properties, concerning their complexity and their relation with one-dimension CA following the rules 102 or 60. These CA can be seen as particular cases of deterministic synchronous BN whose dependency graph is a line graph, where the homogeneous evolution operator is given by the mentioned Boolean functions. Although there are some previous results relating Sierpinsky triangle and some linear CA, the interest and novelty of these results are due to the relations of these Boolean topics with binomial and binary sequences established in this paper, and their possible applications in cryptography.

In the paper "Effects of Anti-modularity and Multiscale Influence in Random Boolean Networks", Escobar et al., the authors extend work on modular BN [2] to measure novel aspects and define anti-modularity. Modular networks enhance the potential criticality of BN, while anti-modular BN turned out to be very similar to regular BN. Even when they have a peculiar structure given by anti-modularity, their dynamics resembles that of networks with random structure. A novel multi-scale model was also proposed, where the states of nodes at a higher scale are determined by lower scale BN and vice versa. Results showed that the statistical properties of the dynamics, such as complexity, are determined by the lower scale (upward causation), while the precise dynamics of the networks are determined by the higher scale (downward causation).

In the article "A Multilayer Structure Facilitates the Production of Antifragile Systems in Boolean Network Models" by Kim et al., the authors use Kim's model of multilayer BNs [3], where genetic dynamics are modeled at a lower layer and intercellular signaling at a higher layer, to show that the multilayer structure increases the probability of observing antifragile dynamics. That is, multilayer BN have the potential of benefiting from noise more than random BN. This suggests that a multilayer structure could be useful in different engineering systems.

In his work "Design of Fixed Points in Boolean Networks Using Feedback Vertex Sets and Model Reduction", Kobayashi provides methods for the design of fixed points in Boolean models of gene regulatory networks using model reduction and interaction graphs. His work includes an illustration of theory through a model for apoptosis, taken from Tournier and Chaves (2009) [4].

The article "Properties exploring and information mining in consumer community network: A case of Huawei Pollen Club" by Meng et al. is devoted to exploring properties and mining information in consumer community network. The consumer community network is constructed by Boolean retrieve programming and discussed in the methodology and empirical way based on the community data of Huawei P10/P10 Plus. The authors conclude that consumer community network is the important place that reflects product experiences and facilitates product innovation in future. Manufacturers can promote improvement and innovation of products by exploring effective information on the consumer community network, thus improving the experience level of consumers. On this basis, three strategies to improve information mining in consumer community networks are proposed.

In the paper "A Novel Antifragility Measure Based on Satisfaction and Its Application to Random and Biological Boolean Networks", Pineda et al. propose a general measure of antifragility. Antifragility occurs when a system benefits from perturbations [5]. Exploring random BN with this measure, the authors found that ordered dynamics are the most antifragile. Also, seven biological BNs exhibited antifragility.

The paper "Solutions to All-Colors Problem on Graph Cellular Automata" by Zhang and Chao provides solutions to the *All-Colors Problem*, which is a generalization of the *All-Ones Problem*. They proceed over some classes of graphs, dividing the study into two subproblems: *Strong-All-Colors Problem* and *Weak-All-Colors Problem*. In addition, they introduce a new kind of All-Color Problem, so called *k-Random Weak-All-Colors Problem*, which is interesting due to its applications to both combinatorial number theory and CA theory.

Conflicts of Interest

The editors declare that they have no conflicts of interest regarding the publication of this special issue.

Acknowledgments

The editors thank all the authors of the papers submitted to this special issue and all the reviewers for their time and effort in conducting the corresponding review processes. Jose C. Valverde was supported by FEDER OP2014-2020 of Castilla-La Mancha (Spain) under the Grant 2019-GRIN-27168 and by the Ministry of Science, Innovation and Universities of Spain under the Grant PGC2018-097198-B-I00. Henning S. Mortveit was partially supported by grants HDTRA1-17-0118 and HDTRA1-11-D-0016-0001. Carlos Gershenson was supported by UNAM's PAPIIT project IN107919. Yongtang Shi was supported by National Natural Science Foundation of China, Natural Science Foundation of Tianjin (No. 17JCQNJC00300), the China-Slovenia bilateral project "Some topics in modern graph theory" (No. 12-6), Open Project Foundation of Intelligent Information Processing Key Laboratory of Shanxi Province (No. CICIP2018005), and

the Fundamental Research Funds for the Central Universities, Nankai University (63191516).

Jose C. Valverde
Henning S. Mortveit
Carlos Gershenson
Yongtang Shi

References

[1] J. A. Aledo, L. G. Diaz, S. Martinez, and J. C. Valverde, "Solution to the predecessors and gardens-of-eden problems for synchronous systems over directed graphs," *Applied Mathematics and Computation*, vol. 370, pp. 22–28, 2019.

[2] R. Poblanno-Balp and C. Gershenson, "Modular random Boolean networks," *Artificial Life*, vol. 17, no. 4, pp. 331–351, 2011.

[3] H. Kim and H. Sayama, "Robustness and evolvability of multilayer gene regulatory networks," in *Proceedings of the 2018 Conference on Artificial Life*, pp. 546-547, MIT Press, 2018.

[4] L. Tournier and M. Chaves, "Uncovering operational interactions in genetic networks using asynchronous Boolean dynamics," *Journal of Theoretical Biology*, vol. 260, no. 2, pp. 196–209, 2009.

[5] N. Taleb, *Antifragile: Things That Gain from Disorder*, Random House, New York, USA, 2012.

Hindawi
Complexity
Volume 2019, Article ID 2783217, 11 pages
https://doi.org/10.1155/2019/2783217

Research Article

A Multilayer Structure Facilitates the Production of Antifragile Systems in Boolean Network Models

Hyobin Kim [1], **Omar K. Pineda** [1,2] **and Carlos Gershenson** [1,3,4]

[1]*Centro de Ciencias de la Complejidad, Universidad Nacional Autónoma de México, 04510 CDMX, Mexico*
[2]*Posgrado en Ciencia e Ingeniería de la Computación, Universidad Nacional Autónoma de México, 04510 CDMX, Mexico*
[3]*Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas, Universidad Nacional Autónoma de México, 04510 CDMX, Mexico*
[4]*ITMO University, St. Petersburg, 199034, Russia*

Correspondence should be addressed to Carlos Gershenson; cgg@unam.mx

Received 27 February 2019; Revised 27 May 2019; Accepted 1 August 2019; Published 9 December 2019

Academic Editor: Diego R. Amancio

Antifragility is a property from which systems are able to resist stress and furthermore benefit from it. Even though antifragile dynamics is found in various real-world complex systems where multiple subsystems interact with each other, the attribute has not been quantitatively explored yet in those complex systems which can be regarded as multilayer networks. Here we study how the multilayer structure affects the antifragility of the whole system. By comparing single-layer and multilayer Boolean networks based on our recently proposed antifragility measure, we found that the multilayer structure facilitated the production of antifragile systems. Our measure and findings will be useful for various applications such as exploring properties of biological systems with multilayer structures and creating more antifragile engineered systems.

1. Introduction

Antifragility is a property from which systems are able to resist stress and furthermore benefit from it [1]. Although the notion of antifragility has been extensively used in many fields like computer science [2–5], transportation [6, 7], engineering [8–10], physics [11], risk analysis [12, 13], and molecular biology [14, 15], a practical quantitative measure of antifragility had not been developed. For that reason, using random Boolean networks (RBNs) and biological BNs, we recently proposed a novel metric that quantifies antifragility [16].

We measured antifragility of BNs based on the change of complexity before and after adding perturbations, in which the BNs were all single-layer networks. However, numerous real-world complex systems are composed of interacting multiple subsystems, which can be regarded as multilayer networks [17, 18]. Here we aim at investigating how the multilayer structure affects the antifragility of the whole system by assessing the antifragility of single-layer and multilayer RBNs and comparing them.

If the multilayer structure has an advantage in gaining antifragility over a single-layer structure, we could utilize the characteristic for a number of areas using BNs [19–27], from understanding properties of biological systems with multilayer structures to designing more antifragile engineered systems.

The rest of this paper is organized as follows. In the section of "Measurement of Antifragility in single-layer and multilayer RBNs", we explain single-layer and multilayer network models, how to calculate their complexity, perturbations to networks, and how to assess the antifragility. In the section "Experiments", specific experimental designs are described. In the section of "Results and Discussion", the results about the antifragility of single-layer/multilayer RBNs and a biological BN are mentioned. The last section summarizes and concludes the paper.

2. Measurement of Antifragility in Single-layer & Multilayer RBNs

2.1. Single-layer and Multilayer Random Boolean Networks. RBNs were suggested as models of gene regulatory

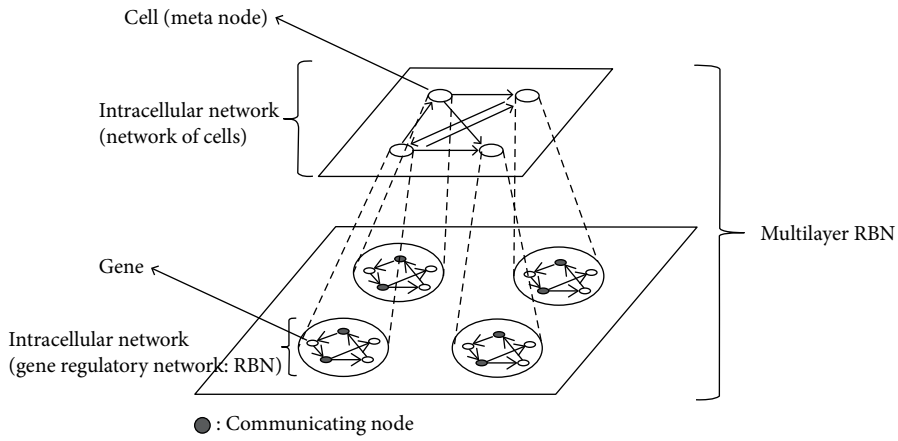


Figure 1: A schematic diagram of a multilayer RBN model. In actual simulations, the number of nodes of an intercellular network was 9, and the number of nodes of an intracellular network was 18.

networks (GRNs) in cells that are present in all known living organisms [28–30]. Although RBNs are highly simplified models, they can greatly explain relevant properties of life and its possibilities. Accordingly, they have been actively used in many fields such as systems biology and artificial life [31–36]. In this study, a single-layer RBN represents a GRN at a single cell level, a multilayer RBN indicates coupled GRNs at a multicellular level.

A RBN is also called NK Boolean network, where N is the number of nodes, and K is the number of input links per node. Here self-links are allowed. In a RBN, the links are randomly arranged, and Boolean functions are randomly assigned to each node as well. Once the topology and Boolean logic rules are determined, they are maintained. Each node represents a gene. The state of a node can have either 0 (off, inhibited) or 1 (on, activated), and it is updated by the states of input nodes and corresponding Boolean functions.

A state space of a RBN is the set of all possible configurations (2^N) of a system including the transitions among them. In the state space, stationary configurations are attractors (point or cyclic), and the others converging into attractors are their basin of attraction. The dynamics of RBNs is divided into ordered, chaotic, or critical regimes by the structure of the state space. The ordered and chaotic regimes indicate phases. The critical regime refers to the phase transition boundary between them. K can change the dynamics of RBNs systematically: ordered for $K = 1$, critical for $K = 2$, and chaotic for $K \geq 3$, on average under internal homogeneity (*i.e.*, probability of being activated or inhibited) $p = 0.5$ [37].

Our multilayer RBN model is composed of two layers: intercellular and intracellular [38, 39]. In an intercellular layer, cells interact with each other and in an intracellular layer, genes interact with each other (Figure 1). All the cells have the same RBNs, and cellular topologies representing interactions between cells keep changing in each simulation. The assumption on such dynamical cellular topology is based on research showing that interacting cells continue to change by cell movements or cell growth [40].

In the multilayer RBN model, the states of all the nodes are simultaneously updated as a whole system. The specific update rules are as follows:

(i) Communicating genes: in each RBN, communicating nodes are assigned for cell-cell interactions, which follow cell signaling in Flann et al.'s model [31]. The state of a communicating node is determined by communicating nodes of neighboring cells. If even one is activated among the communicating nodes of its neighbors, it is activated. Here the neighbors mean source nodes from which the links originate.

(ii) The other genes: the states of the other nodes are determined in the same way as the update of the node states in a single-layer RBN mentioned above.

2.2. Complexity of RBNs. We measure complexity of RBNs based on our previous approach [41, 42] as follows:

$$E_i = -\left(p_0 \log_2 p_0 + p_1 \log_2 p_1\right), \tag{1}$$

$$C = 4 \times \overline{E} \times \left(1 - \overline{E}\right), \tag{2}$$

where E_i is the "emergence" of node i, p_0 (p_1) is the probability of how many times 0 (1) is expressed in node i during T time steps, C ($0 \leq C \leq 1$) is the complexity of the RBN, and $\overline{E}$ ($0 \leq \overline{E} \leq 1$) is average obtained from the emergence values for every node of the network. Specifically, p_0 (p_1) is computed from simulation time $T + 1$ to $2T$ not from 1 to T, which is to obtain p_0 (p_1) in more stable state transitions (*i.e.*, closer to attractors). Figure 2 shows an example calculating complexity of a RBN.

Emergence here means novel information, so it can be measured precisely with Shannon's information entropy (equation (1)). Complexity is conceptually understood as a balance between regularity and change [29]. In equation (2), emergence $\overline{E}$ represents change, and its complement $1 - \overline{E}$ indicates regularity. In our previous study, for regular RBNs it

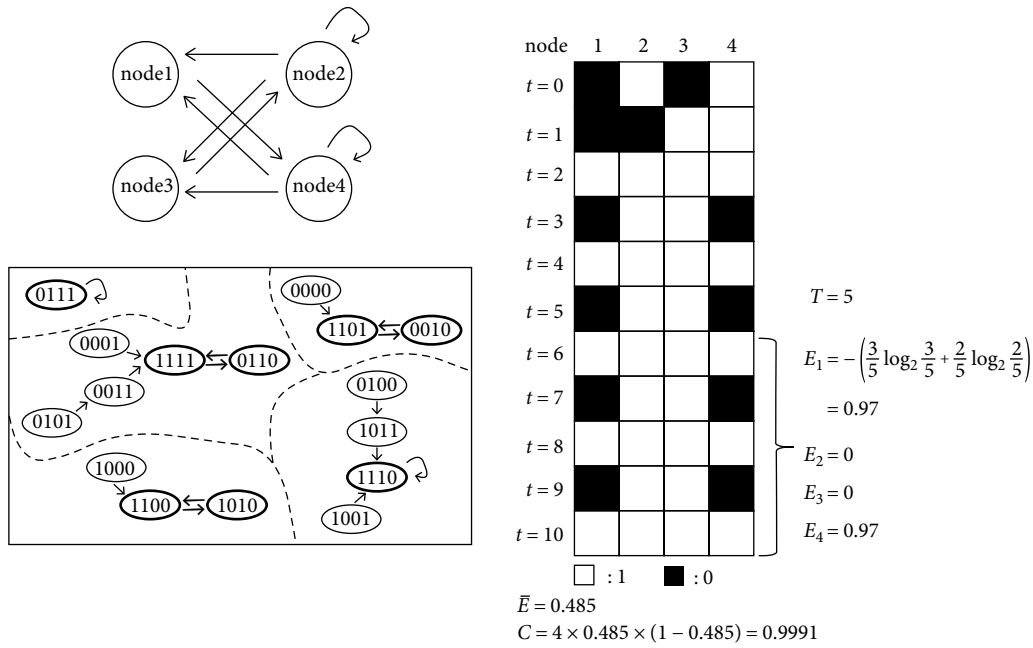


FIGURE 2: An example showing how to calculate complexity. Top-left: a RBN with $N = 4, K = 2$. Bottom-left: state space of the RBN. The state space is composed of $2^4 = 16$ configurations and transitions among them. The configurations with bold outlines are attractors. Dashed lines draw boundaries for each basin of attraction. Right: state transitions and the computation of complexity, based on the emergence of each of the four nodes (columns). 0101 was used as an initial state. The state transitions were obtained from $t = 0$ to $t = 10$.

was maximized at the phase transition, *i.e.,* in the critical regime [41]. Our complexity measure is similar to Galas et al. set complexity [43]. Set complexity, based on Kolmogorov's intrinsic complexity, quantifies the amount of information in a set of objects. Pairs of objects that are maximally redundant or completely random carry negligible information. They calculated the set complexity of trajectories of RBNs and found that the quantity was maximized in critical regime.

To get back to the point about our complexity measure, we can interpret the regularity and the change from an information viewpoint. Regularity enables information to be preserved, and change allows new information to be explored [42]. In the context of RBNs used as GRN models, keeping and changing the node states which point out genetic information can be connected with stability to maintain existing functions and adaptability to flexibly adapt to a new environment.

In equation (2), an optimal balance between regularity and change is achieved at $\overline{E} = 0.5$ ($\overline{E} = 0.5 \rightarrow C = 1$). That is, when either p_0 or p_1 is about 0.89, the complexity has its maximum. [41, 44]. On the contrary, the complexity becomes its minimum when the emergence $\overline{E}$ is 0 or 1 ($\overline{E} = 0$ or $1 \rightarrow C = 0$). It is when only one state is expressed (p_0 or $p_1 = 1; \overline{E} = 0$) or the two states are expressed at the same ratio ($p_0 = p_1 = 0.5; \overline{E} = 1$). The coefficient 4 is added to normalize C to the [0, 1] interval.

2.3. Network Perturbations to RBNs. We perturb RBNs by flipping node states. During $2 \times T$ time steps, we randomly choose X nodes in a RBN consisting of N nodes, and perturb the nodes with frequency O. The perturbations are introduced at $t \bmod O = 0$ (*i.e.*, only when the time step t can be divided by O). For example, $X = 3, O = 5$, and $T = 10$ mean that we randomly choose three nodes of a network at each time step, and flip the node states every five time steps from the initial to 2×10 time steps. The perturbed three nodes are different every five time steps. Then, we calculate fragility based on the states transitions during ten time steps from $t = 11$ to $t = 20$.

To normalize fragility values between -1 and 1, we define the degree of perturbations as follows:

$$\Delta x = \frac{X \times (T/O)}{N \times T}, \tag{3}$$

where $0 \leq \Delta x \leq 1$.

2.4. Antifragility of RBNs. (anti)fragility $\oint$ ($-1 \leq \oint \leq 1$) is defined as follows:

$$\oint = -\Delta\sigma \times \Delta x, \tag{4}$$

where $\Delta\sigma$ is the difference of "satisfaction" from perturbations, and Δx is the degree of perturbations added to a system. The satisfaction σ means the degree of how much agents attain their goal [45]. The satisfaction is contingent on what the defined system is. In this study, each node of the RBN is an agent, and their goal is defined as high complexity. In other words, networks have higher satisfaction when they are closer to criticality. The satisfaction is computed using complexity. However, one can measure the satisfaction using other criteria such as performance and fitness.

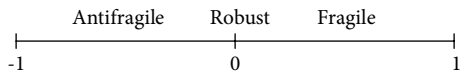
If a system does not get the satisfaction from perturbations and rather is damaged, it means the system is fragile. If the system does not change against perturbations, then it is robust. If the system increases its satisfaction with perturbations, it is antifragile.

In RBNs, the difference of satisfaction is computed based on the difference of complexity before and after adding perturbations. $\Delta\sigma$ is computed by the following equation:

$$\Delta\sigma = C - C_0, \tag{5}$$

where C_0 is complexity of a RBN before perturbations are added to the network, and C is complexity of the RBN after perturbations are introduced to the network. For an original RBN and its perturbed one, the same initial states are applied at $t = 0$. C_0 and C have values between 0 and 1. Thus, $\Delta\sigma$ has values between −1 and 1 ($-1 \leq \Delta\sigma \leq 1$). Regarding Δx, it is described in the section of "Network Perturbations to RBNs".

If the $\oint$ of a RBN has a negative value, the RBN is considered antifragile. If $\oint$ is a positive value, the RBN is fragile. If $\oint$ is close to zero, the RBN is robust.



Based on equation (4), $\oint$ has negative values when C is bigger than C_0, which means that the complexity is increased by perturbations. On the other hand, $\oint$ has positive values when C_0 is larger than C, which indicates that the complexity is reduced due to the perturbations. $\oint$ becomes zero when C is equal to C_0. It represents that the complexity is the same before and after adding the perturbations.

3. Experiments

We conducted three experiments using single-layer & multilayer RBNs and our (anti) fragility measure $\oint$ described as materials and methods in the section above.

(1) Antifragility in multilayer RBNs: we study how $\oint$ of ordered ($K = 1$), critical ($K = 2$), and chaotic ($K = 3, 4$) multilayer RBNs dynamically varies depending on the frequency and size of perturbations, and parameters related to the multilayer structure.

(2) Comparison of antifragility between multilayer and single-layer RBNs: we investigate differences between multilayer and single-layer RBNs by comparing probability of generating antifragile networks.

(3) Comparison of antifragility between multilayer and single-layer *CD4+ T-cell* networks: we examine differences between multilayer and single-layer biological BNs taking a *CD4+ T-cell* network as a real biological example. We compare average values of $\oint$ and probability of generating antifragile networks between them.

Parameter settings for simulations are shown below.

In the case of single-layer RBNs, we produced ordered, critical, and chaotic RBNs. Specifically, 10 different initial states were randomly chosen per a RBN, and then the state transitions from each initial were investigated during $2 \times 400 = 800$ time steps. Using the same initial states, we also looked into the state transitions of the perturbed RBNs. Comparing them during the last 400 time steps, we computed respective $\oint$ from the 10 initial states to obtain their mean. The plots show the values of $\oint$, which are averages from 100 different RBNs for each K.

For multilayer RBNs, we generated three regimes of multilayer networks taking ordered, critical, chaotic RBNs. The topology at an intercellular layer was randomly determined based on the number of links randomly chosen between 1 and 81 (because the number of cells was set to nine, the intercellular network can have a maximum of 81 links.). For an individual RBN, the genes were set to eighteen and the communicating genes were set to six, which is based on the numbers of genes and cell signaling molecules in the real biological system we used (*i.e.*, *CD4+ T-cell*). In the same manner as $\oint$ of single-layer RBNs, $\oint$ of multilayer RBNs was calculated.

For a biological BN, we made use of a *CD4+ T cell* network consisting of 18 nodes. The network for *CD4+ T cell differentiation and plasticity* is modeled to study immune responses controlled by *CD4+ T cells* in terms of factors such as immunological challenges and environmental signals [46]. In the network, there are six cytokines which are cell signaling molecules related to cell-cell communications. We considered the six cytokines communicating nodes in our multilayer network model. For both a single-layer *CD4+ T cell* network and a multilayer *CD4+ T-cell* network, 1000 different initial states were randomly chosen and then the state transitions from each initial were examined during $2 \times 400 = 800$ time steps. Changing the parameters X and O, we computed $\oint$.

For the simulation, the following parameters were used:

(i) Number of genes (N_g).
(ii) Number of in-degrees per node (K).
(iii) Number of cells (N_c).
(iv) Number of communicating genes (C_g).
(v) Number of links of an intercellular network (L_i).
(vi) Number of total genes at a multicellular level ($N_T = N_g \times N_c$).
(vii) Simulation time (T).
(viii) Number of perturbed genes (X).
(ix) Perturbation frequency (O).
(x) Number of different networks
(xi) Number of initial states.

The specific values of parameters follow Table 1. Our simulator was implemented in Java.

4. Results and Discussion

4.1. Antifragility in Multilayer RBNs. Figure 3(a) shows average fragility of multilayer RBNs for $K = 1, 2, 3, 4$

Table 1: Parameters for simulations and their values.

Figure	N_g	K	N_c	C_g	L_i	N_T	T	X	O	# of different networks	# of initial states
3(a)	18	1,2,3,4	9	6	U(1,81)	162	400	80	1..50	100	10
3(b)	18	1,2,3,4	9	6	U(1,81)	162	400	1..162	1	100	10
4(a)	18	1,2,3,4	9	6	U(1,81)	162	400	–	–	100	10
4(b)	18	1,2,3,4	9	6	U(1,81)	162	400	1..162	1	100	10
4(c)	18	1,2,3,4	9	6	U(1,81)	162	400	1..162	1	100	10
5(a)	18	1,2,3,4	9	1..18	U(1,81)	162	400	30	1	100	10
5(b)	18	1,2,3,4	9	6	10..80	162	400	30	1	100	10
6(a)	18	1,2,3,4	9	6	U(1,81)	162	400	1..162	1..5	100	10
6(b)	162	1,2,3,4	1	–	–	162	400	1..162	1..5	100	10
6(c)	18	1,2,3,4	1	–	–	18	400	1..18	1..5	100	10
8(a)	18	1,2,3,4	9	6	U(1,81)	162	400	1..162	1..5	1	1000
8(b)	18	1,2,3,4	1	–	–	18	400	1..18	1..5	1	1000

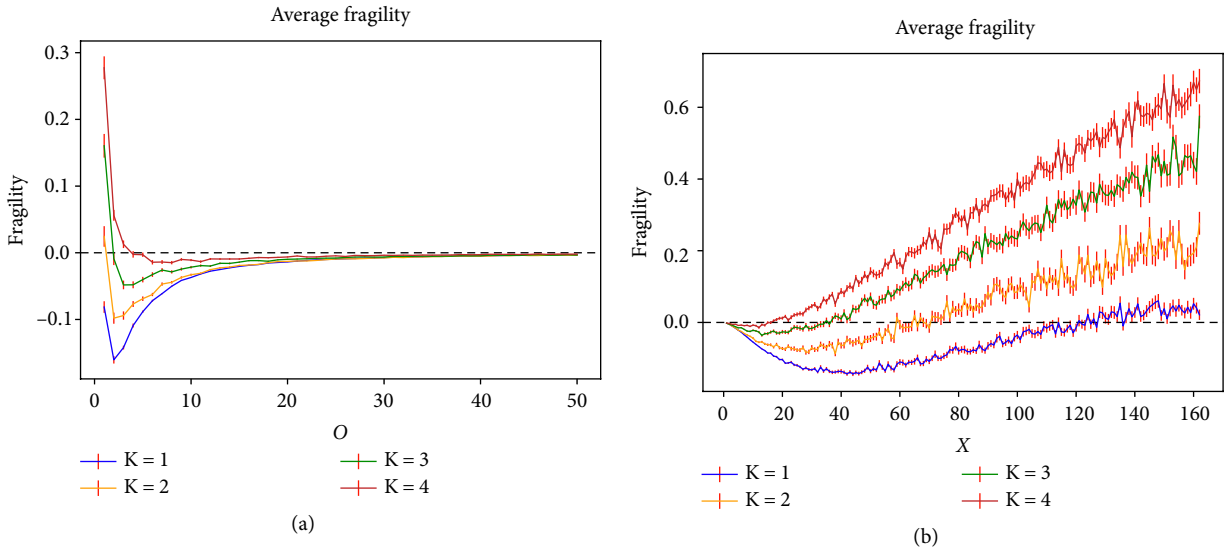


Figure 3: Average fragility of ordered ($K = 1$), critical ($K = 2$) and chaotic ($K = 3, 4$) multilayer RBNs depending on (a) O and (b) X. The error bars represent the standard errors of measurements. For each K, 100 different networks were used. For each network, 10 initial states were randomly chosen. (a) $N_T = 162$, $T = 400$, and $X = 80$. (b) $N_T = 162$, $T = 400$, and $O = 1$.

depending on perturbation frequency O (*i.e.*, the period of adding perturbations) when perturbed node size $X = 80$. With O growing, the antifragile or fragile dynamics of the ordered, critical and chaotic networks changed into robust beyond $O = 30$ even though about half nodes were perturbed ($X = 80$). This result means that the perturbation frequency is more important than the perturbed node size.

Figure 3(b) represents average fragility of multilayer RBNs depending on X when $O = 1$. For all K, there were certain ranges of X for the networks to be antifragile. We also found that the certain ranges of X decreased and antifragility declined as K increased. These findings indicate that "optimal" antifragility results from a moderate level of perturbations, and multilayer RBNs take bigger benefits from perturbations in the order of ordered, critical, and chaotic networks. From Figures 3(a) and 3(b), we can see that maximal antifragility is obtained from a moderate level of perturbations. In addition, it is worth noting that the maximum antifragility varies for different K values.

Figure 4 explains the reason why multilayer RBNs obtained more improved antifragility in the order of ordered, critical, chaotic networks. In Figure 4(a), the complexity before perturbations gradually increased as K got bigger, while in Figure 4(b) the complexity after perturbations increased as K got smaller excluding the early range of X ($1 \leq X \leq 20$). Figure 4(c) shows the difference of complexity before and after perturbations. As seen in the figure, the smaller K was, the larger the difference was. It means that the complexity representing the balance between regularity and change can be improved by perturbations, and the degree of the improvement is much larger for smaller K.

Regarding the complexity, one interesting finding is that the complexity before perturbations (Figure 4(a)) is not consistent with that of previous studies [41, 47, 48]. The existing studies demonstrated that critical RBNs have higher complexity, while our result revealed that chaotic multilayer RBNs have larger complexity. The difference is due to a cellular level. The

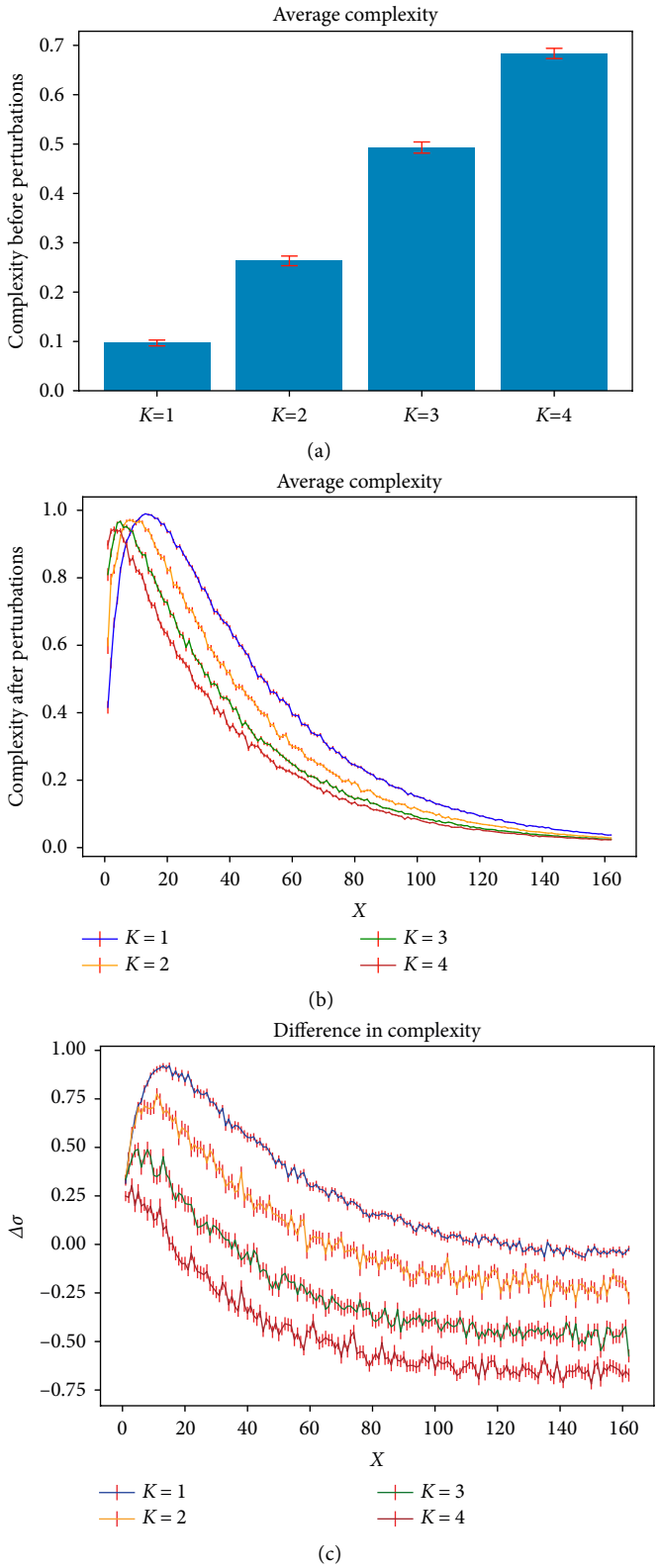


FIGURE 4: Average complexity of ordered, critical, and chaotic multilayer RBNs. (a) Complexity before perturbations. (b) Complexity after perturbations. (c) Difference of complexity before and after perturbations.

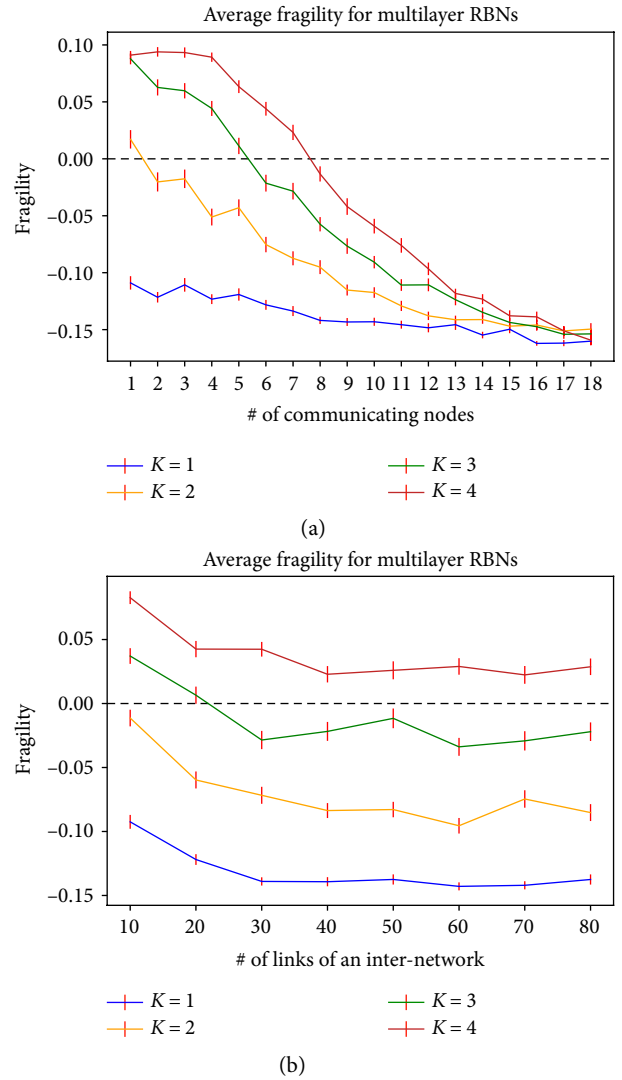


FIGURE 5: Average fragility of ordered, critical, and chaotic multilayer RBNs depending on two parameters related to the multilayer structure. (a) Fragility against the number of communicating nodes. (b) Fragility against the number of links of an intercellular network.

previous studies were performed at a single cell level, and our study was conducted at a multicellular level. The different results between multilayer and single-layer RBNs emphasize the need for research in the context of multicellular settings.

Figure 5 shows average fragility of multilayer RBNs depending on two parameters related to the multilayer structure: the number of communicating genes and the number of links of an intercellular network. In Figure 5(a), as the number of communicating genes increased, multilayer RBNs in all the regimes became antifragile. In Figure 5(b), the fragility values did not change significantly except for the early range as the number of links of an intercellular network increased. These results indicate that the communicating genes have a larger effect on antifragility at a multicellular level. Also, it suggests the possibility that the

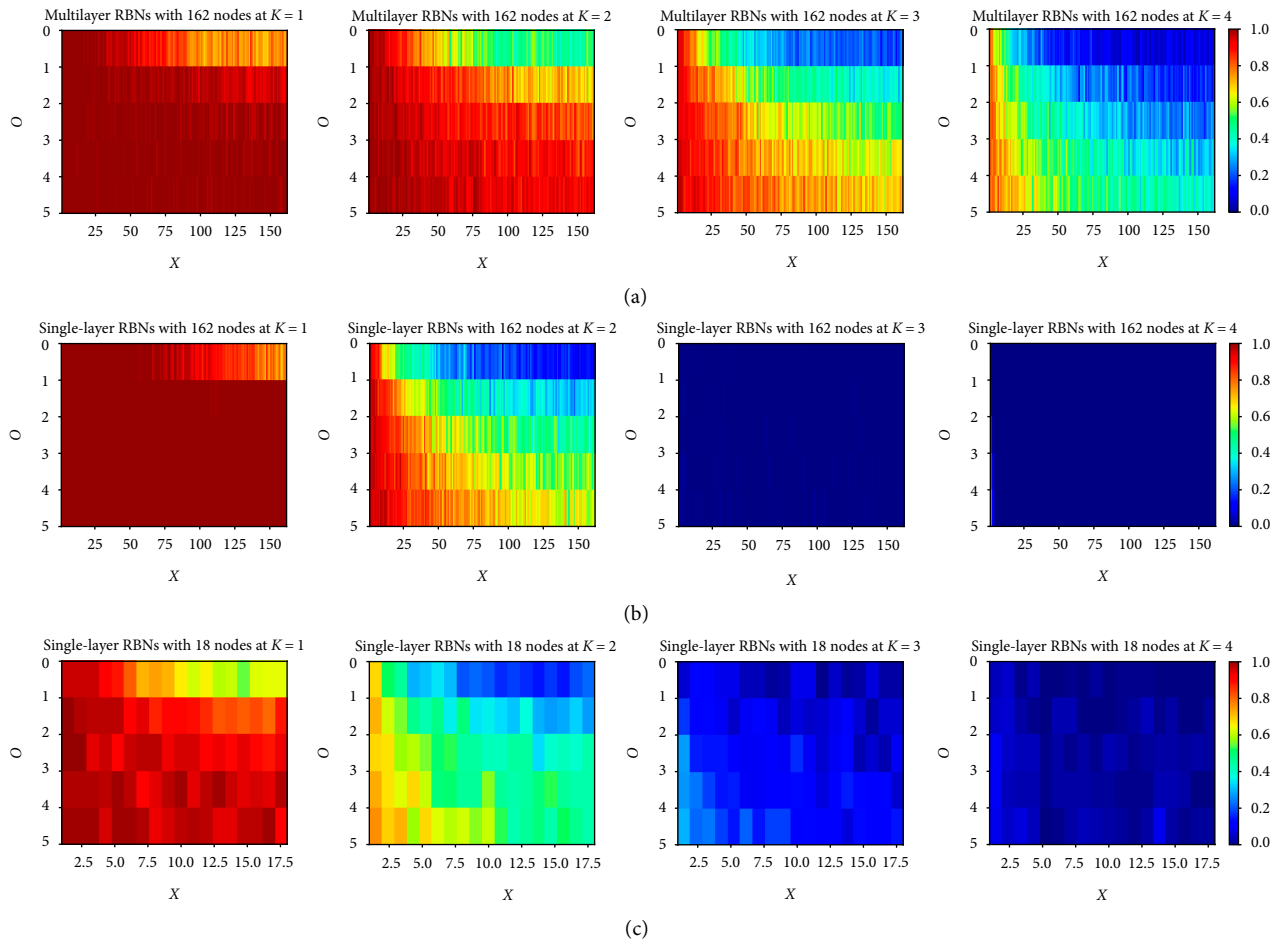


FIGURE 6: Probability of generating antifragile networks depending on X and O for $K = 1, 2, 3, 4$ (from left toward right). The probability is between 0 and 1. In the color bar, blue represents the minimum probability, and red means the maximum value. (a) Probability of producing antifragile networks in multilayer networks ($N_T = 162$). (b) Probability of producing antifragile networks in single-layer networks ($N_T = 162$) with the same state space size as multilayer networks. (c) Probability of producing antifragile networks in single-layer networks ($N_T = 18$) with the same node size as the number of genes in one cell of the multilayer network model.

number of communicating genes representing the degree of interactions between cells might be able to be used as an indicator to estimate the effect of multilayer structure on antifragility.

4.2. Comparison of Antifragility Between Multilayer and Single-layer RBNs. To study how the multilayer structure has an effect on the production of antifragile networks, we calculated probability of how many antifragile networks were generated in multilayer and single-layer RBNs, respectively, and then compared them. Figure 6 shows heat maps representing the probability values in a diverse range of X and O for multilayer and single-layer networks.

Figure 6(a) is the probability in multilayer RBNs with $N_T = 162$. Figure 6(b) is the probability in single-layer RBNs with $N_T = 162$ which have the same state space size as multilayer RBNs (*i.e.*, the state space size $= 2^{162}$). Figure 6(c) is the probability in single-layer RBNs with $N_T = 18$ which have the same node size as the number of genes in one cell of the multilayer network model.

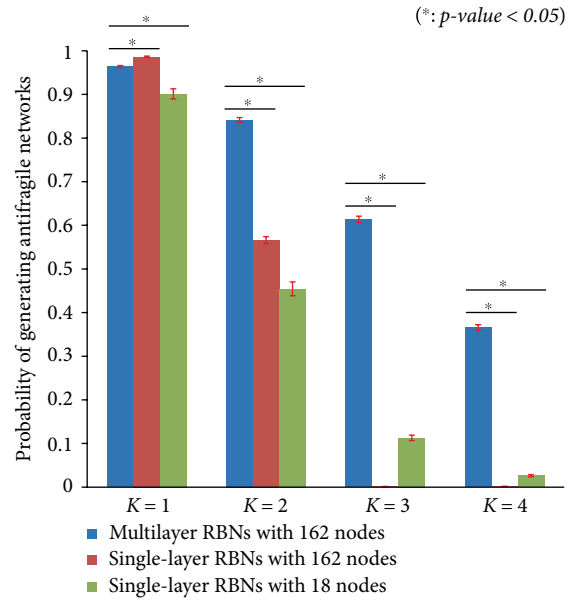


FIGURE 7: Comparison between multilayer and single-layer RBNs based on the probability of generating antifragile networks.

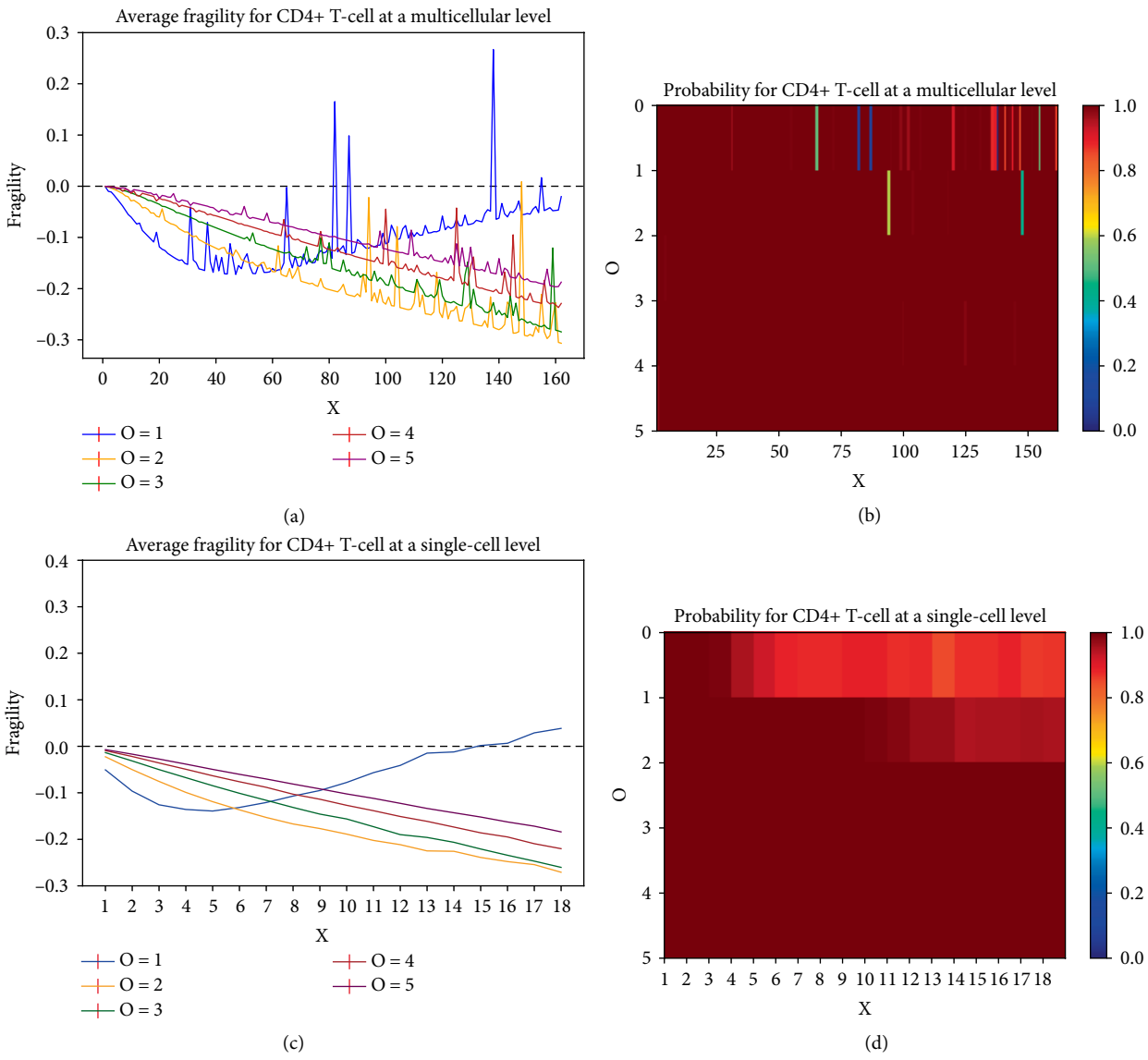


FIGURE 8: Average fragility and the probability of generating antifragile networks in multilayer and single-layer *CD4+ T-cell* networks. (a) Fragility for *CD4+ T-cell* at a multicellular level. (b) Probability for *CD4+ T-cell* at a multicellular level. (c) Fragility for *CD4+ T-cell* at a single-cell level. (d) Probability for *CD4+ T-cell* at a single-cell level.

For all the cases of (a), (b), and (c) in Figure 6, we found that they had similar trends that the smaller K was the more frequently antifragile networks were produced. It means that the trends of antifragile dynamics at a single-cell level are still maintained at a multicellular level. Also, as the perturbed node size increased and the period of adding perturbations became shorter, the probability of generating antifragile networks decreased overall. It is worth noticing that, especially for large K values, there is not much difference between the single-layer RBNs independently of their size (Figures 6(b) and 6(c)). On the other hand, multilayer RBNs are clearly more antifragile (Figure 6(a)).

Figure 7 shows the comparison of the probability acquired from Figure 6 based on a two-sample t-test. Firstly, to investigate the difference of antifragility between multicellular and single-cell systems with the same system size, we compared multilayer RBNs with 162 nodes and single-layer RBNs with 162 nodes. As seen in the figure, the multilayer networks produced antifragile networks more frequently at $K = 2, 3, 4$. In the case of $K = 1$, the single-layer networks had higher probability, but the values were not so different.

Secondly, to examine the difference of antifragility between a multicellular system and its component, we compared multilayer RBNs with 162 nodes and single-layer RBNs with 18 nodes. We found that the multilayer networks generated antifragile networks with higher probability for all K. The findings from Figure 7 indicate that the multilayer structure helps to produce the greater number of antifragile networks, especially for larger K values.

4.3. Comparison of Antifragility Between Multilayer and Single-layer CD4+ T-cell Networks. Using a *CD4+ T-cell* network related to the immune system as an example of biological

systems, we calculated average fragility and the probability of generating antifragile networks for an individual *CD4+ T-cell* network and coupled *CD4+ T-cell* networks. As seen in Figures 8(a) and 8(c), the general tendency of antifragile dynamics in the *CD4+ T-cell* network at a multicellular level was practically the same as the tendency at a single-cell level.

We compared multilayer and single-layer *CD4+ T-cell* networks based on the probability values in Figures 8(b) and 8(d). We found that multilayer *CD4+ T-cell* networks generated antifragile networks with modestly higher probability on a two-sample t-test (*i.e.*, multicellular level: 99.26% > single-cell level: 97.74% under p-value <0.05). From a biological viewpoint, these results suggest that the properties of biological systems might be enhanced in the structure of interacting multiple subsystems.

When compared to multilayer and single-layer RBNs, the *CD4+ T-cell* network showed similar antifragile dynamics to the dynamics of multilayer and single-layer networks at $K = 1$. From this, we can infer that the *CD4+ T-cell* network may be ordered. This can be understood because immune cells probably not only have a variable environment, but actually have evolved to thrive on it. The finding on the ordered dynamics of the *CD4+ T-cell* network is consistent with many research findings exhibiting that gene regulatory networks of biological systems have ordered or critical dynamics [49–52].

5. Conclusions

In this study, applying our (anti)fragility measure to multilayer and single-layer BNs, we studied how the dynamics of the networks varies depending on relevant parameters, and how the multilayer structure affects the antifragility of the whole system. We found that systems showed different dynamics depending on the degree of perturbations and the degree of interaction between system components: fragile, robust, or antifragile. Also, we found that the multilayer structure facilitated the production of antifragile systems. Probably this is related to the modular structure of multilayer RBNs [33], although further studies should be made.

The findings can be utilized for various applications such as systems biology and bio-inspired engineering. For example, our results may be helpful to figure out dynamical characteristics of multicellular organisms. Also, we could create engineered systems with an increased antifragility based on the fact that system properties can vary from fragile through robust to antifragile dynamics depending on the size and frequency of perturbations, and the number of communicating nodes.

Our study has a few limitations. Firstly, our multilayer RBN model is the one where identical RBNs are randomly coupled. However, there are many other systems where different subsystems are connected to each other and they communicate in a certain way. Secondly, we used only one biological example in explaining the dynamical behaviors of biological systems. To obtain more generalized findings on antifragility, we plan to develop different kinds of multilayer network models, explore them, and use various biological systems.

Data Availability

Our source code and data are available at https://github.com/bin20005/antifragility.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Funding

This research was partially supported by CONACYT and DGAPA, UNAM.

Acknowledgments

We are grateful to Dario Alatorre for useful comments and discussions.

References

[1] N. N. Taleb, *Antifragile: Things that Gain from Disorder*, Random House New York, NY, USA, 2012.

[2] C. A. Ramirez and M. Itoh, "An initial approach towards the implementation of human error identification services for antifragile systems," in *Proceedings of the SICE Annual Conference (SICE)*, IEEE, pp. 2031–2036, Sapporo, Japan, 2014.

[3] A. Abid, M. T. Khemakhem, S. Marzouk, M. B. Jemaa, T. Monteil, and K. Drira, "Toward antifragile cloud computing infrastructures," *Procedia Computer Science*, vol. 32, pp. 850–855, 2014.

[4] M. Monperrus, "Principles of antifragile software," in *Proceedings of Companion to the First International Conference on the Art, Science and Engineering of Programming*, ACM, Brussels, Belgium, 2017.

[5] L. Guang, E. Nigussie, J. Plosila, and H. Tenhunen, "Positioning antifragility for clouds on public infrastructures," *Procedia Computer Science*, vol. 32, pp. 856–861, 2014.

[6] J. S. Levin, S. P. Brodfuehrer, and W. M. Kroshl, "Detecting antifragile decisions and models lessons from a conceptual analysis model of service life extension of aging vehicles," in *Proceedings of 2014 IEEE International Systems Conference*, pp. 285–292, IEEE, Ottawa, ON, Canada, 2014.

[7] R. Isted, "The use of antifragility heuristics in transport planning," in *Proceedings of Australian Institute of Traffic Planning and Management (AITPM) National Conference*, TRB, Adelaide, South Australia, Australia, 2014.

[8] K. H. Jones, "Engineering antifragile systems: a change in design philosophy," *Procedia Computer Science*, vol. 32, pp. 870–875, 2014.

[9] M. Lichtman, M. T. Vondal, T. C. Clancy, and J. H. Reed, "Antifragile communications," *IEEE Systems Journal*, vol. 12, no. 1, pp. 659–670, 2018.

[10] E. Verhulst, "Applying systems and safety engineering principles for antifragility," *Procedia Computer Science*, vol. 32, pp. 842–849, 2014.

[11] A. Naji, M. Ghodrat, H. Komaie-Moghaddam, and R. Podgornik, "Asymmetric coulomb fluids at randomly charged dielectric interfaces: antifragility, overcharging and charge inversion," *The Journal of Chemical Physics*, vol. 141, no. 17, p. 174704, 2014.

[12] J. Derbyshire and G. Wright, "Preparing for the future: development of an 'antifragile' methodology that complements scenario planning by omitting causation," *Technological Forecasting and Social Change*, vol. 82, pp. 215–225, 2014.

[13] T. Aven, "The concept of antifragility and its implications for the practice of risk analysis," *Risk Analysis*, vol. 35, no. 3, pp. 476–483, 2015.

[14] M. Grube, L. Muggia, and C. Gostinčar, "Niches and Adaptations Of Polyextremotolerant Black Fungi," *Polyextremophiles*, Springer, Netherlands, 2013.

[15] A. Danchin, P. M. Binder, and S. Noria, "Antifragility and tinkering in biology (and in business) flexibility provides an efficient epigenetic way to manage risk," *Genes*, vol. 2, no. 4, pp. 998–1016, 2011.

[16] O. K. Pineda, H. Kim, and C. Gershenson, "A novel antifragility measure based on satisfaction and its application to random and biological boolean networks," *Complexity*, vol. 2019, Article ID 3728621, 10 pages, 2019.

[17] M. Kivelä, A. Arenas, M. Barthelemy, J. P. Gleeson, Y. Moreno, and M. A. Porter, "Multilayer networks," *Journal of Complex Networks*, vol. 2, no. 3, pp. 203–271, 2014.

[18] S. Boccaletti, G. Bianconi, R. Criado et al., "The structure and dynamics of multilayer networks," *Physics Reports*, vol. 544, no. 1, pp. 1–122, 2014.

[19] R. Serra, M. Villani, A. Barbieri, S. A. Kauffman, and A. Colacci, "On the dynamics of random Boolean networks subject to noise: attractors, ergodic sets and cell types," *Journal of Theoretical Biology*, vol. 265, no. 2, pp. 185–193, 2010.

[20] A. Bauer, T. L. Jackson, Y. Jiang, and T. Rohlf, "Receptor cross-talk in angiogenesis: mapping environmental cues to cell phenotype using a stochastic, boolean signaling network model," *Journal of Theoretical Biology*, vol. 264, no. 3, pp. 838–846, 2010.

[21] M. K. Morris, J. Saez-Rodriguez, P. K. Sorger, and D. A. Lauffenburger, "Logic-based models for the analysis of cell signaling networks," *Biochemistry*, vol. 49, no. 15, pp. 3216–3224, 2010.

[22] R. Albert and H. G. Othmer, "The topology of the regulatory interactions predicts the expression pattern of the segment polarity genes in *Drosophila melanogaster*," *Journal of Theoretical Biology*, vol. 223, no. 1, pp. 1–18, 2003.

[23] F. Li, T. Long, Y. Lu, Q. Ouyang, and C. Tang, "The yeast cell-cycle network is robustly designed," *Proceedings of the National Academy of Sciences*, vol. 101, no. 14, pp. 4781–4786, 2004.

[24] M. Paczuski, K. E. Bassler, and A. Corral, "Self-organized networks of competing boolean agents," *Physical Review Letters*, vol. 84, no. 14, pp. 3185–3188, 2000.

[25] M. E. J. Newman, "Spread of epidemic disease on networks," *Physical Review E*, vol. 66, no. 1, Article ID 016128, 2002.

[26] P. Rämö, S. A. Kauffman, J. Kesseli, and O. Yli-Harja, "Measures for information propagation in boolean networks," *Physica D: Nonlinear Phenomena*, vol. 227, no. 1, pp. 100–104, 2007.

[27] A. Roli, M. Manfroni, C. Pinciroli, and M. Birattari, "On the design of boolean network robots," in *Proceedings of European Conference on the Applications of Evolutionary Computation*, pp. 43–52, Springer, Berlin, Heidelberg, 2011.

[28] S. A. Kauffman, "Metabolic stability and epigenesis in randomly constructed genetic nets," *Journal of Theoretical Biology*, vol. 22, no. 3, pp. 437–467, 1969.

[29] S. A. Kauffman, *The Origins of Order Self-Organization and Selection in Evolution*, Oxford University Press, NY, USA, 1993.

[30] S. A. Kauffman, *At Home in the Universe: The Search for the Laws of Self-Organization and Complexity*, Oxford University Press, NY, USA, 1996.

[31] N. S. Flann, H. Mohamadlou, and G. J. Podgorski, "Kolmogorov complexity of epithelial pattern formation: the role of regulatory network configuration," *BioSystems*, vol. 112, no. 2, pp. 131–138, 2013.

[32] H. Mohamadlou, G. J. Podgorski, and N. S. Flann, "Modular genetic regulatory networks increase organization during pattern formation," *BioSystems*, vol. 146, pp. 77–84, 2016.

[33] R. Poblanno-Balp and C. Gershenson, "Modular random Boolean networks," *Artificial Life*, vol. 17, no. 4, pp. 331–351, 2011.

[34] C. Gershenson, S. A. Kauffman, and I. Shmulevich, "The role of redundancy in the robustness of random boolean networks," in *Proceedings of the Tenth International Conference on the Simulation and Synthesis of Living Systems (ALIFE X)*, pp. 35–42, Indiana University, Bloomington, Indiana, United States, MIT Press, 2005.

[35] H. Kim and H. Sayama, "The role of criticality of gene regulatory networks in morphogenesis," *IEEE Transactions on Cognitive and Developmental Systems*, 2018.

[36] H. Kim and H. Sayama, "How criticality of gene regulatory networks affects the resulting morphogenesis under genetic perturbations," *Artificial Life*, vol. 24, no. 2, pp. 85–105, 2018.

[37] B. Derrida and Y. Pomeau, "Random networks of automata: a simple annealed approximation," *Europhysics Letters*, vol. 1, no. 2, pp. 45–49, 1986.

[38] H. Kim and H. Sayama, "Robustness and evolvability of multilayer gene regulatory networks," in *Proceedings of Conference on Artificial Life (ALIFE 2018)*, pp. 546–547, MIT Press, Tokyo, Japan, 2018.

[39] H. Kim, "*The role of criticality of gene regulatory networks on emergent properties of biological systems,*" Graduate Dissertations and Theses, Article ID 93, 2018.

[40] M. D. Jackson, S. Duran-Nebreda, and G. W. Bassel, "Network-based approaches to quantify multicellular development," *Journal of The Royal Society Interface*, vol. 14, no. 135, Article ID 20170484, 2017.

[41] N. Fernández, C. Maldonado, and C. Gershenson, "Information measures of complexity, emergence, self-organizationhomeostasis, and autopoiesis," *Guided Self-Organization: Inception*, Springer, M. Prokopenko, Ed., 2014.

[42] G. Santamaría-Bonfil, C. Gershenson, and N. Fernández, "A package for measuring emergence, self-organization, and complexity based on Shannon entropy," *Frontiers in Robotics and AI*, vol. 4, 10 pages, 2017.

[43] D. J. Galas, M. Nykter, G. W. Carter, N. D. Price, and I. Shmulevich, "Biological information as set-based complexity," *IEEE Transactions on Information Theory*, vol. 56, no. 2, pp. 667–677, 2010.

[44] C. Gershenson and N. Fernández, "Complexity and information: measuring emergence, self-organization, and homeostasis at multiple scales," *Complexity*, vol. 18, no. 2, pp. 29–44, 2012.

[45] C. Gershenson, "The sigma profile: a formal tool to study organization and its evolution at multiple scales," *Complexity*, vol. 16, no. 5, pp. 37–44, 2011.

[46] M. E. Martinez-Sanchez, L. Mendoza, C. Villarreal, and E. R. Alvarez-Buylla, "A minimal regulatory network of extrinsic and intrinsic factors recovers observed patterns of CD4+ T cell differentiation and plasticity," *PLOS Computational Biology*, vol. 11, no. 6, 2015.

[47] C. Gershenson, "Introduction to random boolean networks," in *Workshop and Tutorial Proceedings of Ninth International Conference on the Simulation and Synthesis of Living Systems (ALife IX)*, pp. 160–173, Boston, United States, 2004.

[48] S. A. Kauffman, *Investigations*, Oxford University Press, 2000.

[49] I. Shmulevich, S. A. Kauffman, and M. Aldana, "Eukaryotic cells are dynamically ordered or critical but not chaotic," *Proceedings of the National Academy of Sciences*, vol. 102, pp. 13439–13444, 2005.

[50] E. Balleza, E. R. Alvarez-Buylla, A. Chaos, S. Kauffman, I. Shmulevich, and M. Aldana, "Critical dynamics in genetic regulatory networks: examples from four kingdoms," *PLOS ONE*, vol. 3, no. 6, 2008.

[51] M. Villani, L. La Rocca, S. A. Kauffman, and R. Serra, "Dynamical criticality in gene regulatory networks," *Complexity*, vol. 2018, Article ID 5980636, 14 pages, 2018.

[52] B. C. Daniels, H. Kim, D. Moore et al., "Criticality distinguishes the ensemble of biological regulatory networks," *Physical Review Letters*, vol. 121, no. 13, p. 138102, 2018.

Hindawi
Complexity
Volume 2019, Article ID 8209146, 14 pages
https://doi.org/10.1155/2019/8209146

WILEY | Hindawi

Research Article

Effects of Antimodularity and Multiscale Influence in Random Boolean Networks

Luis A. Escobar,[1,2] Hyobin Kim,[2] and Carlos Gershenson[2,3,4]

[1] *Posgrado en Ciencia e Ingeniería de la Computación, Universidad Nacional Autónoma de México, 04510 CDMX, Mexico*
[2] *Centro de Ciencias de la Complejidad, Universidad Nacional Autónoma de México, 04510 CDMX, Mexico*
[3] *Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas, Universidad Nacional Autónoma de México, 04510 CDMX, Mexico*
[4] *ITMO University, St. Petersburg 199034, Russia*

Correspondence should be addressed to Carlos Gershenson; cgg@unam.mx

Received 1 March 2019; Revised 12 May 2019; Accepted 21 May 2019; Published 13 June 2019

Academic Editor: Lucia Valentina Gambuzza

We investigate the effects of modularity, antimodularity, and multiscale influence on random Boolean networks (RBNs). On the one hand, we produced modular, antimodular, and standard RBNs and compared them to identify how antimodularity affects the dynamical behaviors of RBNs. We found that the antimodular networks showed similar dynamics to the standard networks. Confirming previous results, modular networks had more complex dynamics. On the other hand, we generated multilayer RBNs where there are different RBNs in the nodes of a higher scale RBN. We observed the dynamics of micro- and macronetworks by adjusting parameters at each scale to reveal how the behavior of lower layers affects the behavior of higher layers and vice versa. We found that the statistical properties of macro-RBNs were changed by the parameters of micro-RBNs, but not the other way around. However, the precise patterns of networks were dominated by the macro-RBNs. In other words, for statistical properties only upward causation was relevant, while for the detailed dynamics downward causation was prevalent.

1. Introduction

The structure of a network can significantly alter its function or behavior. Modularity is a structural property prevalent in many systems, where elements within modules have more connections among themselves than with other elements [1, 2]. Extrapolating from the notion of modularity, we can define antimodularity when elements in subsets have less connections among themselves than with other elements. Modularity has been extensively studied. It is related to the complexity of the systems and their capacity to evolve and adapt, but at the same time to resist perturbations in a robust fashion [3–7]. It maintains robustness through isolation of modules and allows faster adaptation via altering the connections among the modules. On the other hand, what would be the functional properties of antimodular networks?

In addition to modularity, another relevant property of complex systems is the causality between scales in a multilayer structure. It has attracted much attention in systems and computational biology how changes at lower layers affect system properties at upper layers and vice versa [8–10]. Gene therapy is an example showing "upward" causality, where the therapy at the genetic level cascades to an effect on upper levels such as tissues and organs. An example of "downward" causation would be the behavioral choices of an animal selecting a particular mate, which will partly determine which genes will be preserved in future generations.

In this study, we aim to investigate the effects of antimodularity and causality between layers in random Boolean networks (RBNs). Because RBNs have been used as models to represent dynamics of gene regulatory networks [11–13], our research could help better understand the characteristics of many living organisms showing modularity/antimodularity and causality between upper and lower levels [6, 9, 14].

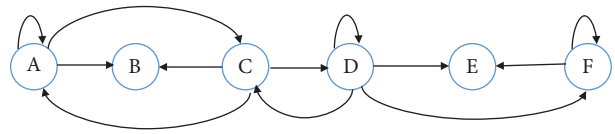


Figure 1: An example RBN ($N = 6$, $K = 2$). The nodes and links are represented by circles and arrows, respectively.

	A	B	C	D	E	F
A	0	1	0	1	0	0
B	0	1	0	0	0	1
C	1	1	0	0	0	0
D	0	0	1	0	1	0
E	0	1	1	0	0	0
F	0	1	0	1	0	0

	A	B	C	D	E	F
A	1	1	0	0	0	0
B	0	1	0	0	0	1
C	1	0	1	0	0	0
D	0	0	1	1	0	0
E	0	0	0	0	1	1
F	0	0	0	1	0	1

	A	B	C	D	E	F
A	0	0	1	1	0	0
B	0	1	0	0	0	1
C	1	1	0	0	0	0
D	0	0	1	0	1	0
E	0	1	1	0	0	0
F	0	1	0	0	0	1

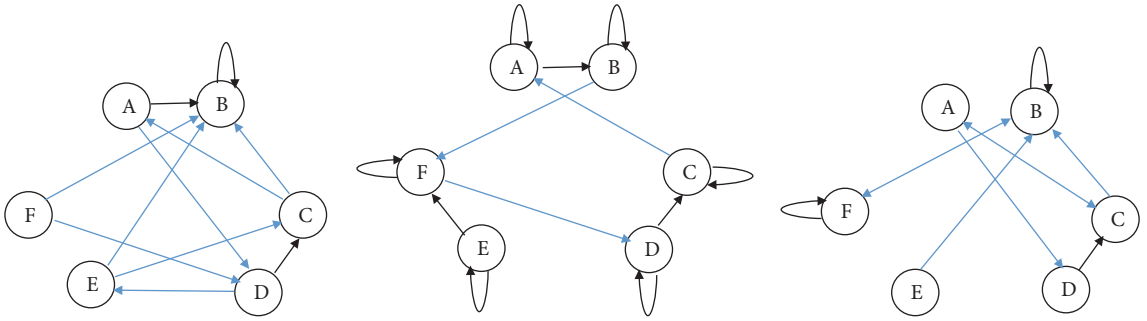


Figure 2: Adjacency matrixes and networks of standard, modular, and antimodular RBNs. All the networks consist of 6 nodes ($N = 6$) and 12 links. In the network, each node has two input nodes on average ($K = 2$). In the adjacency matrixes, blue means that there is a link between corresponding nodes, and yellow represents that there is no link in the defined module. (a) A standard RBN. The links are randomly distributed. (b) A modular RBN with three modules (AB, CD, and EF). There are 9 intramodular links and 3 intermodular links. (c) An antimodular RBN. The same modules as (b) are assumed. There are 3 intramodular links and 9 intermodular links.

2. Materials and Methods

2.1. Random Boolean Networks. RBNs were suggested as gene regulatory network models by Kauffman [11–13]. A RBN is composed of N nodes and K connections per node (or K input nodes). An example RBN is shown in Figure 1. The nodes of the network can only have states of zero (off, inhibited) or one (on, activated). In RBNs, Boolean functions and topologies are randomly arranged. The states of the nodes are determined by the states of input nodes and the logical rules randomly assigned to each node.

2.2. Modularity/Antimodularity of RBNs. A module is a set of nodes that are more densely connected to each other than to nodes from other modules [14, 15]. Based on this definition, we produce modular, antimodular, and standard RBNs. Specifically, centering on diagonal elements in the adjacency matrix of a RBN, we define modules and then arrange links inside and outside the modules. Here we only consider modules which have the same size. Depending on node sizes, the number of modules to be defined in a network can be different. For example, in Figure 2, there are four possible homogeneous modules: 1 (i.e., ABCDEF), 2 (i.e., ABC, DEF), 3 (i.e., AB, CD, EF), and 6 (i.e., A, B, C, D, E, F), although the first and last are trivial, representing the network and node levels, respectively.

The three types of RBNs are as follows:

(i) Standard RBNs: regardless of modules, links are randomly distributed in the adjacency matrix. Figure 2(a) shows an example of a standard RBN.

(ii) Modular RBNs: a greater number of links are randomly placed inside the modules than outside the modules with probability p_m. Figure 2(b) represents an example of a modular RBN with three modules, where each module consists of two nodes.

(iii) Antimodular RBNs: contrary to modular RBNs, a larger number of links are distributed outside the

	A	B	C	D	E	F
A	1	0	0	1	0	0
B	1	1	0	0	0	0
C	1	0	1	0	0	0
D	0	0	1	1	0	0
E	0	0	0	0	1	1
F	0	0	0	0	1	1

Potential intramodular links= 12
Potential intermodular links = 24
Intramodular links = 10
Intermodular links = 2
Nodes outside the Module = 2
$\%i = \frac{10}{12} = 0.83$ $\%o = \frac{2}{24} = 0.083\%$
$M = \frac{\%i}{\%o}$ $M = \frac{0.83\%}{0.083\%} = 10$

(a)

	A	B	C	D	E
A	1	1	0	0	0
B	1	0	0	1	0
C	0	0	1	0	1
D	1	0	1	0	0
E	0	0	0	1	1

Potential intramodular links= 13
Potential intermodular links = 12
Intramodular links = 8
Intermodular links = 2
Nodes outside the Module = 2
$\%i = \frac{8}{13} = 0.61$ $\%o = \frac{2}{12} = 0.16\%$
$M = \frac{\%i}{\%o}$ $M = \frac{0.61\%}{0.16\%} = 3.81$

(b)

Figure 3: Examples showing how to calculate the modularity M. Yellow boxes indicate modules. (a) Modularity with homogeneous modules. (b) Modularity with nonhomogeneous modules.

defined modules than inside the modules with probability p_a. Figure 2(c) refers to an example of an antimodular RBN.

In this study, we consider arbitrarily $p_m = p_a = 0.9$.

We quantitatively measure modularity M as follows:

$$M = \frac{\%i}{\%o} \quad while\ \%o > 0, \tag{1}$$

where $\%i$ is the percentage of possible links within the defined modules (intermodular) and $\%o$ is the percentage of possible links outside the modules (intramodular). If there are no connections outside the modules, that is to say $\%o = 0$, it is considered that the network is infinitely modular, because modules are completely isolated. Figure 3(a) presents an example showing how to calculate the modularity. Antimodularity $\overline{M}$ is defined by the following equation:

$$\overline{M} = \frac{1}{M}. \tag{2}$$

To check the realism of the modules we define and the modularity measure, we compare our modular/antimodular RBNs with biological Boolean networks. In the case of biological networks, we find their modules using Fortunato's community detection algorithm [16], which allows the detection of homogeneous and nonhomogeneous modules. Figure 3(b) is an example showing how to compute the modularity in nonhomogeneous modules.

2.3. Multilayer RBNs. To study the causality between upper and lower levels, we developed multilayer RBN model where there are different RBNs in the nodes of a higher scale RBN. We call the meta RBN *macronetwork* and the RBNs in the nodes of the macronetwork *micronetwork*. The micronetwork consists of disconnected RBNs, although they may interact through the macronetwork (the M of the micronetwork would be infinite). In our model, the node states of the macronetwork and micronetworks are updated through the interactions between them. Specifically, the following processes are repeated:

(1) Random initial states are assigned to micronetworks (Figure 4(a)).

(2) The node states of the micronetworks are updated at the microlevel during t time steps (Figure 4(b)).

(3) The node states of the macronetwork are determined based on the node states of micronetworks at time t. If the node states of the micronetwork in the node of the macronetwork have a larger number of 1s (0s), the node state of the macronetwork becomes 1(0). In other words, the state of the nodes of the macronetwork is determined by the majority of the nodes of each lower scale RBN. (Figure 4(c)).

(4) The node states of the macronetwork are updated during T time steps. The node states of the macronetwork at time T determine the node states of each RBN at the lower scale. If the node state of the macronetwork is 1(0), the states of all its lower scale nodes become 1(0) (Figure 4(d)).

2.4. Experiments. To investigate the effect of modularity/antimodularity, we performed 100 independent simulation runs for each parameter combination. For each simulation, we generated modular, antimodular, and standard RBNs composed of 240 nodes. Because the number of nodes is 240, all the possible numbers of modules to be defined in the network are 2, 3, 4, 5, 6, 8, 10, 12, 15, 16, 20, 24, 30, 40, 48, 60, 80, 120, and 240. Varying K from 1 to 7, we measured modularity, the number of attractors, the average length of attractors, and complexity for all the cases of modules.

Here complexity was calculated based on our previous approach [17, 18]. Complexity C ($0 \le C \le 1$) is defined as follows:

$$\begin{aligned} E_i &= -\left(p_0 \log_2 p_0 + p_1 \log_2 p_1\right) \\ C &= 4 \times \overline{E} \times \left(1 - \overline{E}\right) \end{aligned} \tag{3}$$

where E_i is the emergence of node i and $p_{0(1)}$ is the probability that the state of the node is 0 (1). C is the complexity of the network. It is calculated based on average ($0 \le \overline{E} \le 1$) of the emergence values for all the nodes. 4 is added to normalize the values of C between 0 and 1.

Figure 5 shows how complexity is affected by the number of nodes of a network. Although the number of nodes increases, the networks have the a similar trend: complexity

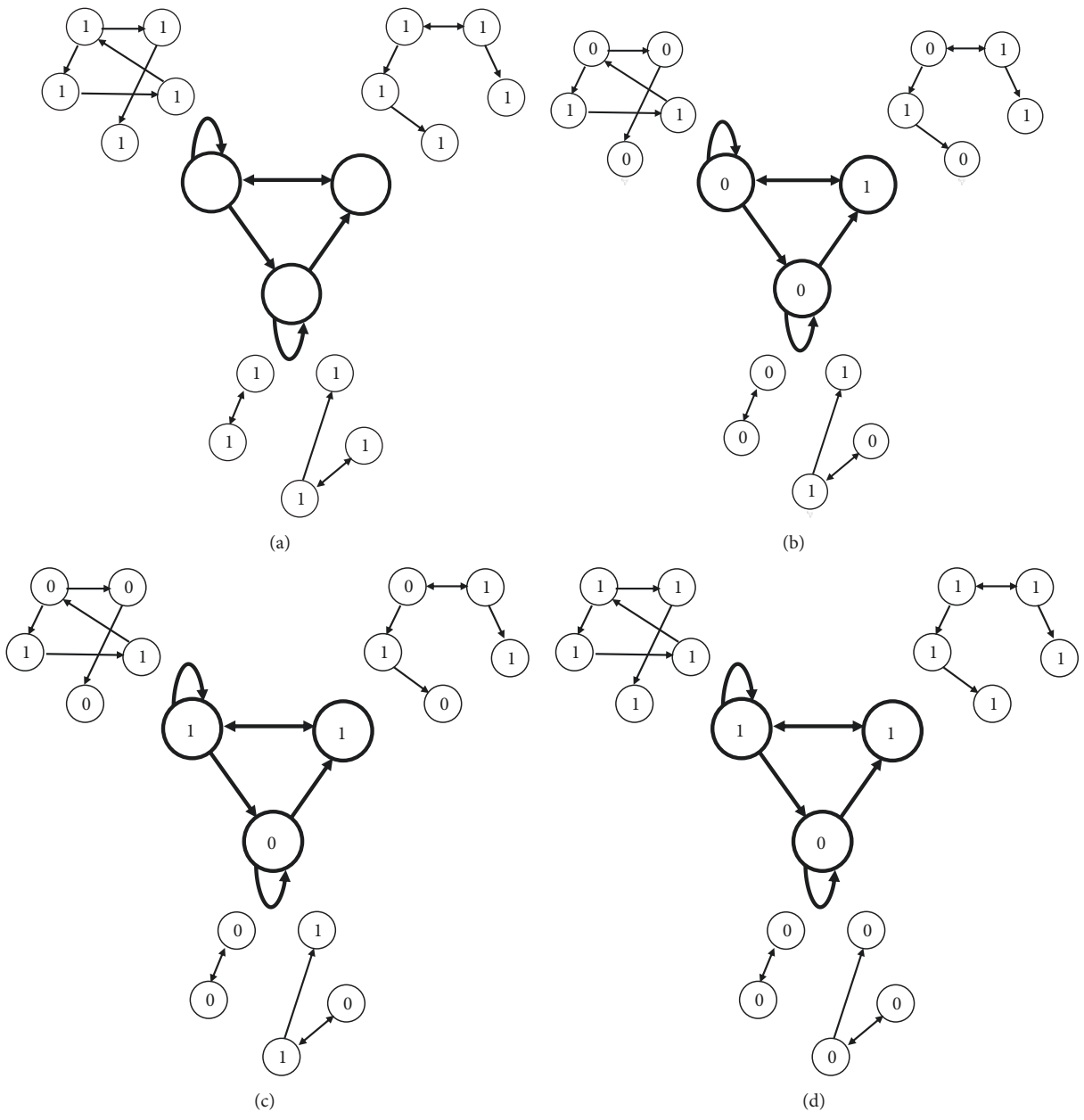


Figure 4: Schematic diagrams illustrating the updates of the node states in a multilayer RBN model.

increases until reaching the maximum at $K = 2.3$ and after the peak, it decreases monotonically with a growing K. However, the precise complexity values do change with the network size, probably due to a finite size effect, which is clearly seen around $K = 2.3$.

To examine the causality between micro- and macronetworks, we conducted 55 independent simulation runs. For each simulation, we produced a multilayer RBN model where each micronetwork has 125 nodes, and the macronetwork has 55 nodes. In the simulation, T of the macronetwork was set to 3. T of the micronetworks was set to 1, 2, 3, 5, and 10. Changing K from 1 to 7, we measured complexity of micro- and macro-RBNs.

3. Results and Discussion

Figure 6 shows modularity, numbers of attractors, complexity, and lengths of attractors for modular RBNs. In a first set of experiments, as K increased, the modularity decreased. In the three plots of modularity, when the modules are more (i.e., the size of the modules is smaller), the modularity drops overwhelmingly as the connections (K) increase. This is because a greater number of links end up being placed outside the modules.

To illustrate how links are distributed in modular networks, we present a few simple examples in Figure 7. The figure represents link distributions in the modules depending on K and module size. As mentioned in the Introduction, K is

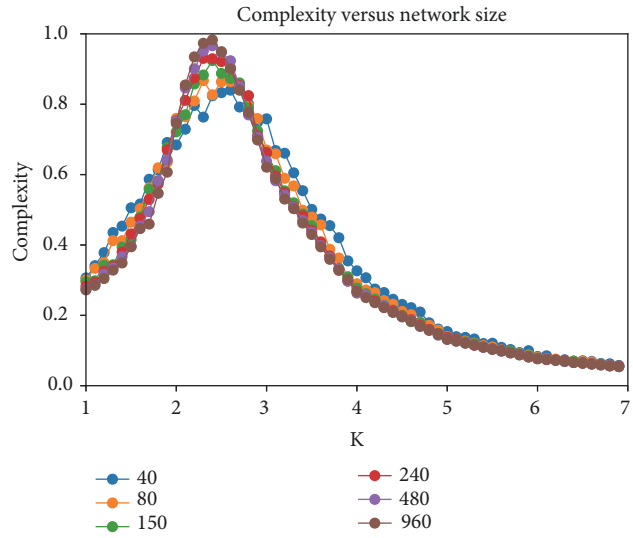


Figure 5: Complexity for different number of nodes, varying average connectivity K.

the number of connections per node. Thus, in the examples, we assume that K connections are distributed per row in the modules. If K is larger than the module size, the rest of the connections are placed outside the modules, which causes the decrease of modularity. As seen in the figure, as the module size gets smaller from 4 to 1 (i.e., the number of modules increases), a tendency can be identified.

In a second set of experiments, the number of attractors is limited to a maximum number of 1,000. In other words, if 1,000 attractors are found, the search for attractors is halted. As seen in the figure, the numbers of attractors found were zero at $K > 3$, which resulted from very long transients and attractors caused by chaotic dynamics of RBNs. Meanwhile, the attractors increased as the number of modules grew at $K = 2$. In the case of 80 and 120 modules, the maximum number of 1,000 attractors was found. However, when the number of modules was 240, the numbers of attractors dramatically decreased.

In a third set of experiments, the complexity gradually increased until reaching their peaks at $K = 2.5$, which is expected as complexity is maximized at the phase transition in standard RBNs; i.e., it reflects criticality. Since then the values started to decrease regardless of the number of modules for the most cases. However, the complexity was affected by the number of modules for a few cases. When the numbers of modules were 40, 48, and 60, the complexity values were the largest, especially in regions where chaotic dynamics would be expected. When the numbers were 120 and 240, the values decreased. This confirmed our previous results [4], where we saw that modularity extends the critical regime (with a high complexity) towards the chaotic region.

In another set of experiments, the lengths of attractors were the highest at $K = 2$ but converged to zero when K was larger than 3. Except for 120 and 240 modules, the more modules a network had, the more increased the lengths of attractors were. When the modules were 120 and 240, the values were decreased.

Figure 8 shows characteristic examples of state transitions depending on the number of modules. As seen in the figure, the states of the network with 240 modules converged to the attractor faster than the states of the network with two modules.

Figure 9 represents numbers of attractors, complexity, and lengths of attractors for antimodular RBNs. The numbers of attractors gradually increased and reached the peaks at $K = 2$ as the number of modules grew (Figure 9(a)). Since then, the values decreased and started to converge into 1 from $K = 3$.

For the complexity, the values had little variation against the number of modules (Figure 9(b)). That is, modularity did not affect complexity in antimodular networks. The complexity increased until reaching the maximum at $K = 2.3$. After that, it decreased monotonically as K grew. The lengths of attractors were not affected by the number of modules (Figure 9(c)). The values increased until reaching their peaks at $K = 2$. After that, it started to decrease and became zero for K larger than 3. Figure 10 shows the complexity of standard and antimodular RBNs. The complexity of antimodular RBNs was very similar to that of standard networks.

To compare the results acquired from our modular/antimodular RBNs and the properties of biological networks, we collected seven biological networks from CellCollective.org [19–25]. Figure 11 shows the adjacency matrix and modularity of one biological network related to treatment of infections. In the adjacency matrix, blue and red boxes indicate nonhomogeneous modules detected by Fortunato's algorithm.

Figure 12 shows the complexity and the modularity of seven biological networks. To study the correlation between complexity and modularity, we calculated Pearson correlation coefficient ($r = 0.4764$). Figure 13 illustrates the correlation graphically. We found that the complexity and the modularity have a moderate positive correlation. This result

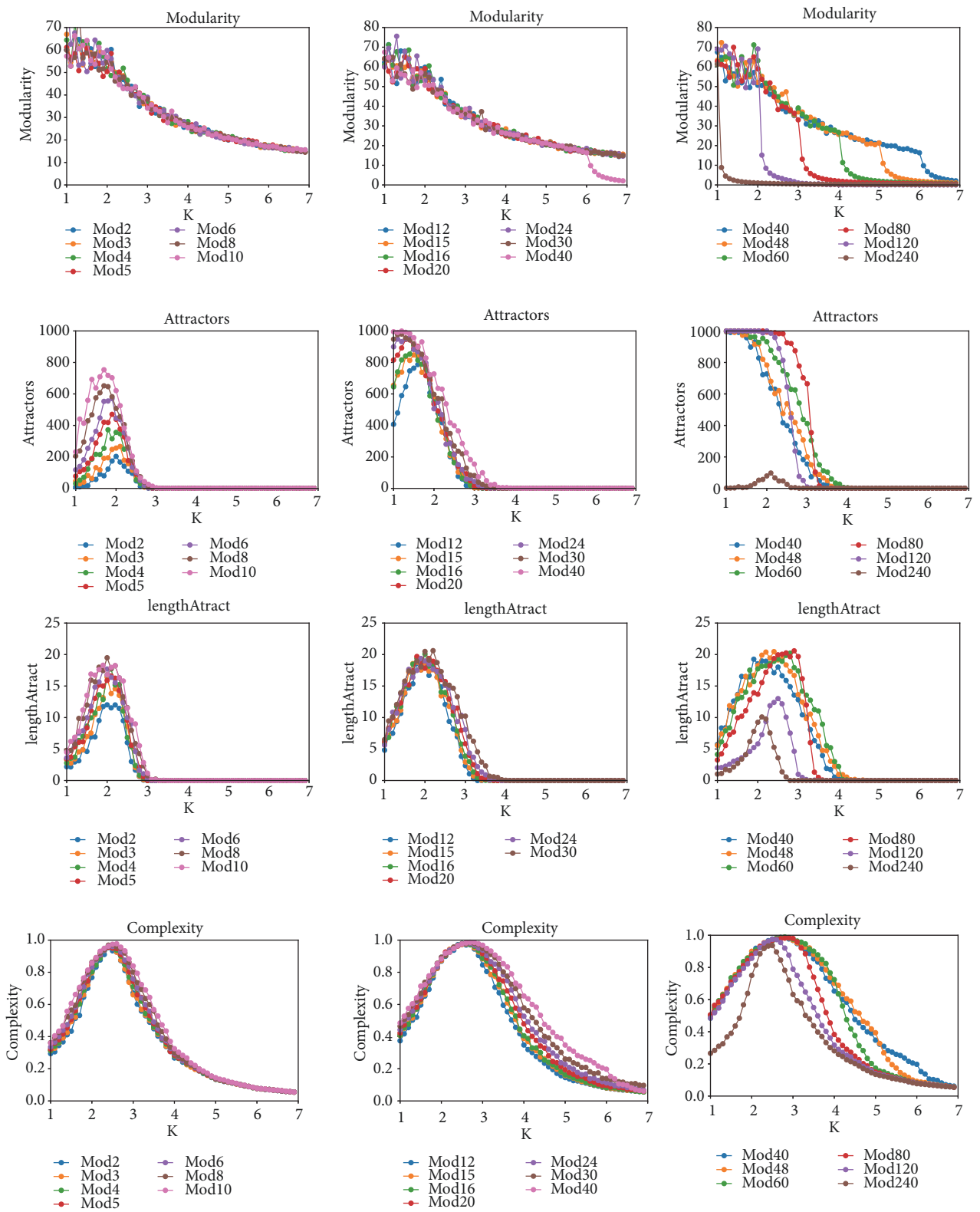


Figure 6: Modularity, numbers of attractors, average length of attractors, and complexity for modular RBNs.

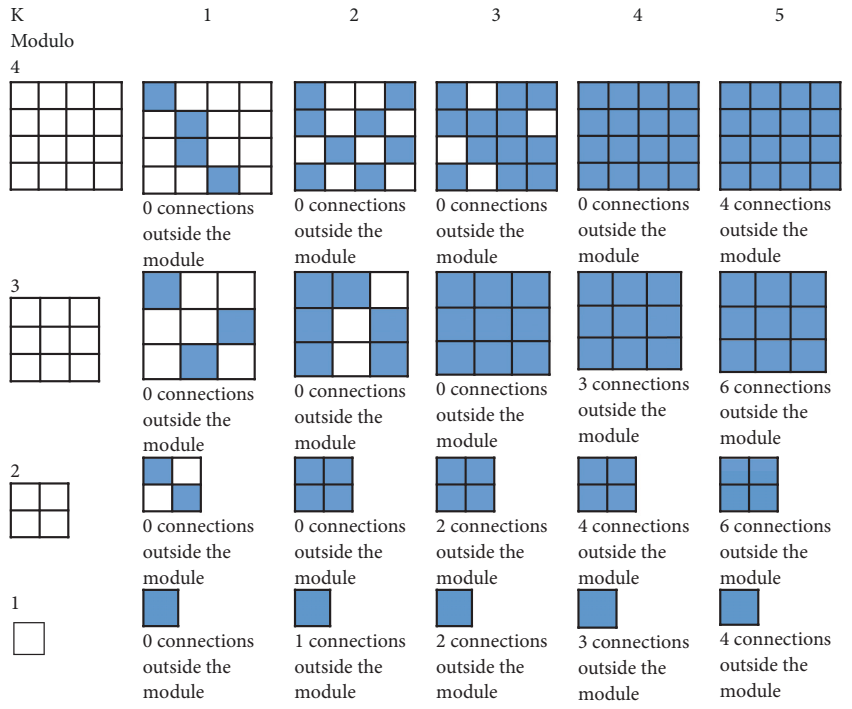


Figure 7: Link distributions depending on K and a module size.

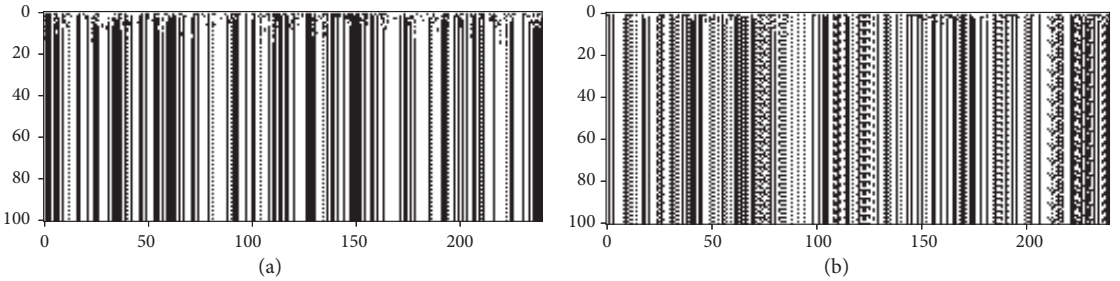


Figure 8: State transitions of networks with $N = 240$ and $K = 2$ depending on the number of modules. The states were calculated during 100 timesteps. Time moves downward, columns represent nodes, and colors show states (white represents 0 and black represents 1). (a) Network with 2 modules. (b) Network with 240 modules.

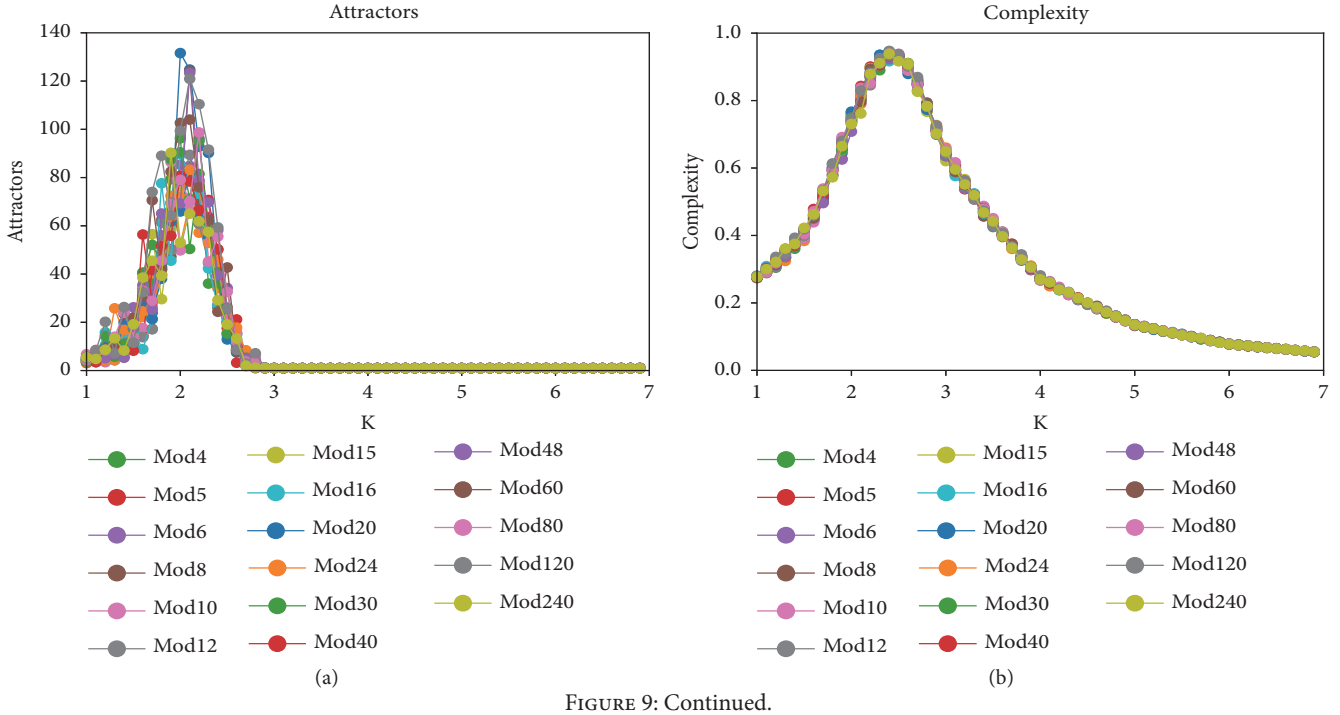
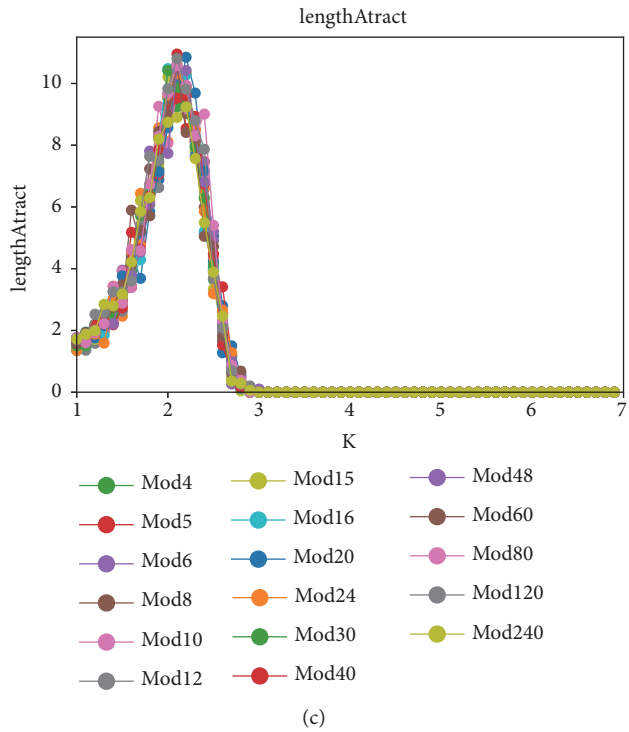


Figure 9: Continued.



(c)

Figure 9: Numbers of attractors, complexity, and lengths of attractors for antimodular RBNs.

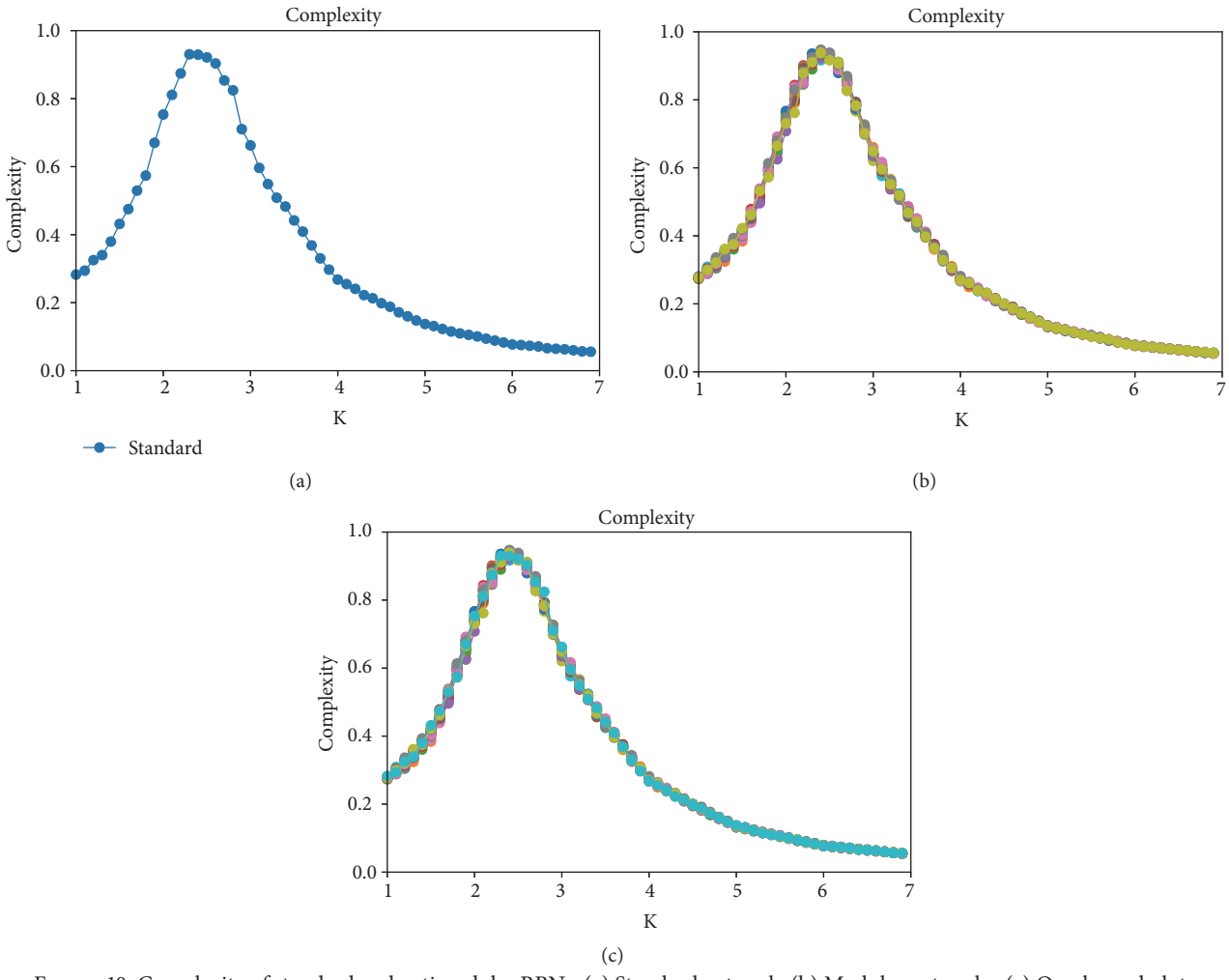


Figure 10: Complexity of standard and antimodular RBNs. (a) Standard network. (b) Modular networks. (c) Overlapped plots.

	Akkermansia	Blautia	Mollicutes	Clostridium	Coprobacillus	Enterobacteri	Enterococcus	Lachnospirac	Lachnospirac	Barnesiella	Other	Clindamycin
Akkermansia	0	0	0	0	1	0	0	0	0	0	0	0
Blautia	0	1	0	0	1	0	1	0	0	0	0	0
Mollicutes	0	0	1	0	0	0	0	0	0	0	0	0
Clostridiumdi	0	0	0	1	0	0	0	0	0	1	0	0
Coprobacillus	0	0	0	0	0	0	0	0	0	0	0	0
Enterobacteri	0	0	0	0	0	1	0	0	0	0	0	0
Enterococcus	0	1	1	1	1	1	0	0	0	0	0	0
Lachnospirac	0	0	0	0	0	0	0	1	1	0	1	0
Lachnospirac	0	0	0	0	0	0	0	1	1	0	1	0
Barnesiella	0	0	0	0	0	0	0	1	1	0	1	1
Other	0	0	0	0	0	0	0	1	1	0	1	1
Clindamycin	0	0	0	0	0	0	0	0	0	0	0	1

Potential intramodular links= 75

Potential intermodular links = 70

Intramodular links = 27

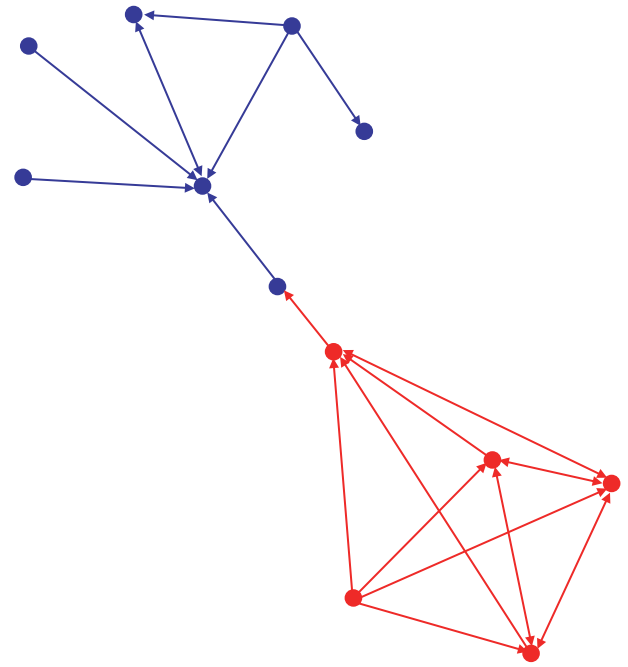
Intermodular links =1

Nodes outside the Module = 1

$$\%i = \frac{27}{75} = 0.36 \qquad \%o = \frac{1}{70} = 0.014\%$$

$$M = \frac{\%i}{\%o} \qquad M = \frac{0.36\%}{0.014\%} = 2.57$$

(a)



(b)

Figure 11: Adjacency matrix and modularity of a biological network related to infection treatment.

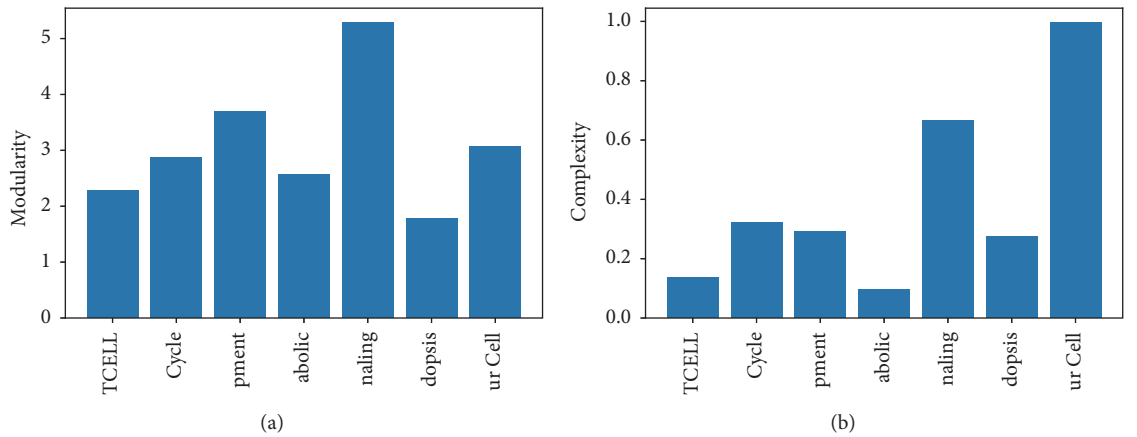


Figure 12: Modularity and complexity of seven biological networks.

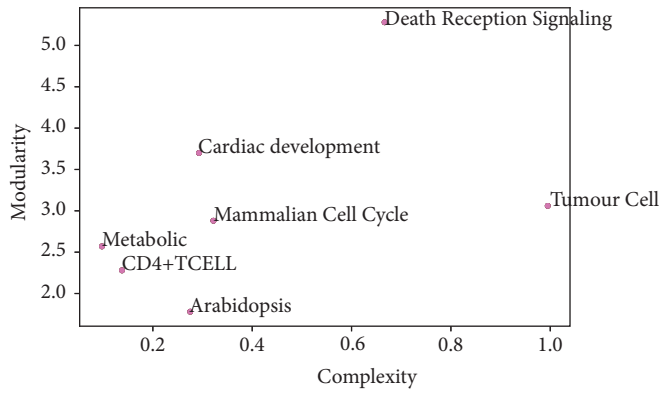


Figure 13: Complexity versus modularity for seven biological networks.

supports our finding that the number of modules has an impact on complexity in the modular RBNs.

Figure 14 shows heat maps on the complexity of micro- and macro-RBNs. When K of the macronetwork (macro-K) was larger than 4, its complexity was changed by K of the micronetworks (micro-K). For micro-K smaller than 2, the complexity was close to zero. However, as the value of K increased, the complexity became higher. This trend stayed the same although t (i.e., the period for the update of the node states of the micronetworks) was varied. In other words, the dynamics of the lower scale affected the higher scale only when ordered dynamics were present at the lower scale: they inhibited the expected chaotic dynamics of the higher scale.

On the contrary, the complexity of the micronetworks was not influenced by macro-K but mostly depended on the micro-K. The complexity of micronetworks increased until micro-K became 2. Since then, the value decreased as micro-K got bigger. This trend was manifested for larger t. Thus, downward causation was not detected in this study for complexity, or at least an effect of the macronetwork on the micronetwork was not detected.

As another experiment for multiscale influence, an analysis of the attractors was performed. What can be seen in Figure 15 is that the attractors of micro- and macronetworks are similar but for the micronetwork are shorter. This result indicates that the attractors of the macronetwork dominate the dynamics of the micronetwork.

Let us recall that an attractor is a set of states that are repeated. If a timestep of the macronetwork does not produce a change in a particular node, this will force its micronetwork to back into a state of only ones or zeroes. Thus, a regular pattern will be observed even if the micronetwork is chaotic. In other words, without the macroinfluence, it would be actually in a transient.

Even if there is a change in the node of the macronetwork, this would push its micronetwork to back into one of two possible states. Thus, on average micronetworks' attractors are shorter because macronodes that do not change force the same initial state over the micronetwork, while the macronetwork can have longer attractors composed of the combinations of the microattractors, as shown in Figure 16.

4. Conclusions

Our computational experiments showed that modularity increases the "diversity" of the network dynamics. That is, there are more and longer attractors as the number of modules increases. For the antimodularity, the increase of the modules does not influence the length or number of attractors.

Networks with a median number of modules extend the complexity of RBNs. All networks have a "critical" region where complexity is high (around 2.3 for standard networks due to a finite size effects, as the theoretical phase transition occurs at $K = 2$ at the thermodynamic limit, i.e., for infinite networks). Networks with very few or a lot of modules have topologies which are more similar to standard topologies, while a median number of modules implies a structure that allows for a better balance of robustness and adaptability, which is precisely a characteristic of standard RBNs with a high complexity. We can conclude that modularity "extends" criticality in RBNs, since our measure of complexity is maximal at the critical region of parameters (not only in RBNs).

Antimodular RBNs showed similar dynamics as standard RBNs. We can infer the reason from Figure 17. In the figure, adjacency matrixes of standard, modular, and antimodular RBNs are presented. In the adjacency matrixes of the standard network and the antimodular one, a greater number of links are distributed outside the diagonal modules. Because the antimodular network has relatively larger space for the distribution of links in the adjacency matrix, especially as the network size increases, it is more similar to the standard network where the links are randomly arranged compared to the modular network.

For the causality between micro- and macro-RBNs, the complexity of the macronetwork was varied by K of the micronetworks when K of the macronetwork was larger than 4. This trend was maintained irrespective of the period for the update of the node states in the micronetworks. Meanwhile, the complexity of the micronetworks was only determined by their own structure. This suggests that there is no downward effect between the two scales in the complexity analysis.

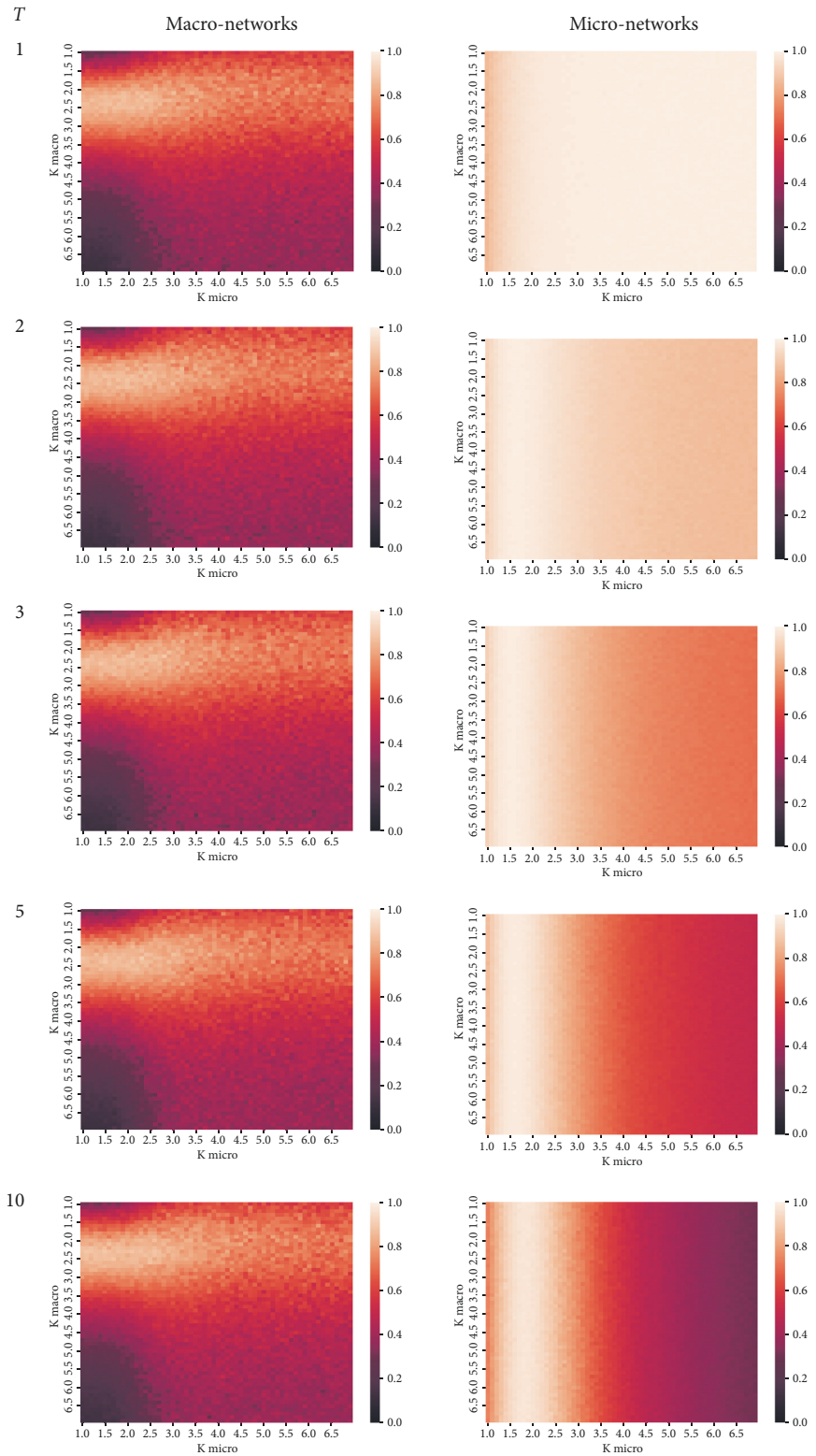


Figure 14: Heat maps of the complexity of micro- and macro-RBNs.

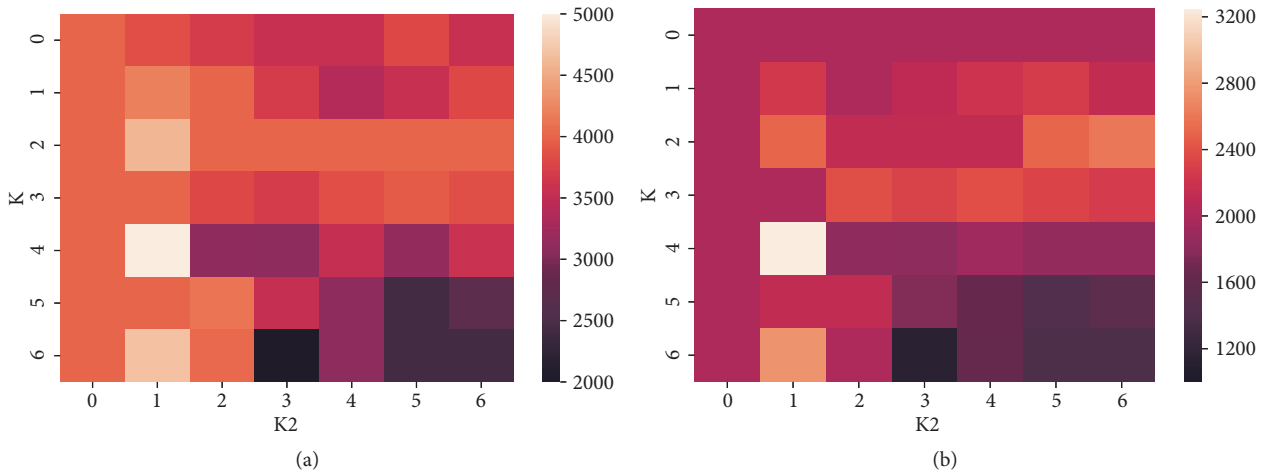


Figure 15: Heat maps showing the numbers of attractors depending on micro- and macro-K. (a) Macronetwork. (b) Micronetwork.

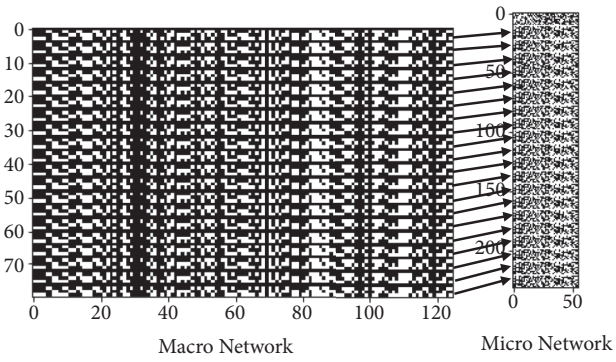


Figure 16: Temporal dynamics (time moves downward, columns represent nodes, and colors represent states) showing how an attractor of the macronetwork generates an attractor in an chaotic micronetwork.

However, the study of the attractors for the multiscale networks showed that the attractors in the macronetwork affect how the micronetwork behaves. Our complexity measure is based on Shannon's information, and thus it is a statistical approximation of the dynamics of a network. In other words, it does not distinguish the precise arrangement of bits and just focuses on their probability distribution. For example, a random bit sequence has the same information (maximal) and complexity (minimal) than an ordered string which has precisely one-half of all zeroes and another half of all ones. In our experiments, the statistical properties of the network dynamics (i.e., the complexity) were determined mainly by the lower scale. However, the precise order of the dynamics (i.e., the attractors) was determined mainly by the higher scale.

Our results suggest that studying only the lower scale of systems would be meaningful only if we are interested in certain statistical properties. Nevertheless, if we want to understand and attempt to predict the precise dynamics, we need to study both scales and how they interact. This can be illustrated with the classical example of different arrangements of carbon atoms, which are the same at the lower scale, but depending on their structure at the higher scale (charcoal, diamond, graphene, etc.) they can have very different properties at both scales. Still, studying whether our results for RBNs can be generalized to all phenomena is an ambitious task which is beyond the scope of this paper.

For further study, we plan to investigate a relationship between modularity/antimodularity and diverse topologies such as cyclic or star shapes and multiscale effects of micro- and macronetworks with different structures. Also, we will study how to control the states of modular, antimodular, and multilayer networks so that our research can be applied to more fields. Specifically, using mathematical approaches like the technique of semitensor product (STP) [26–28], we will examine the dynamics of networks and design controllers for them.

Data Availability

Our simulator and data are available at https://github.com/angelEsc/RBN. Biological networks were obtained from https://research.cellcollective.org.

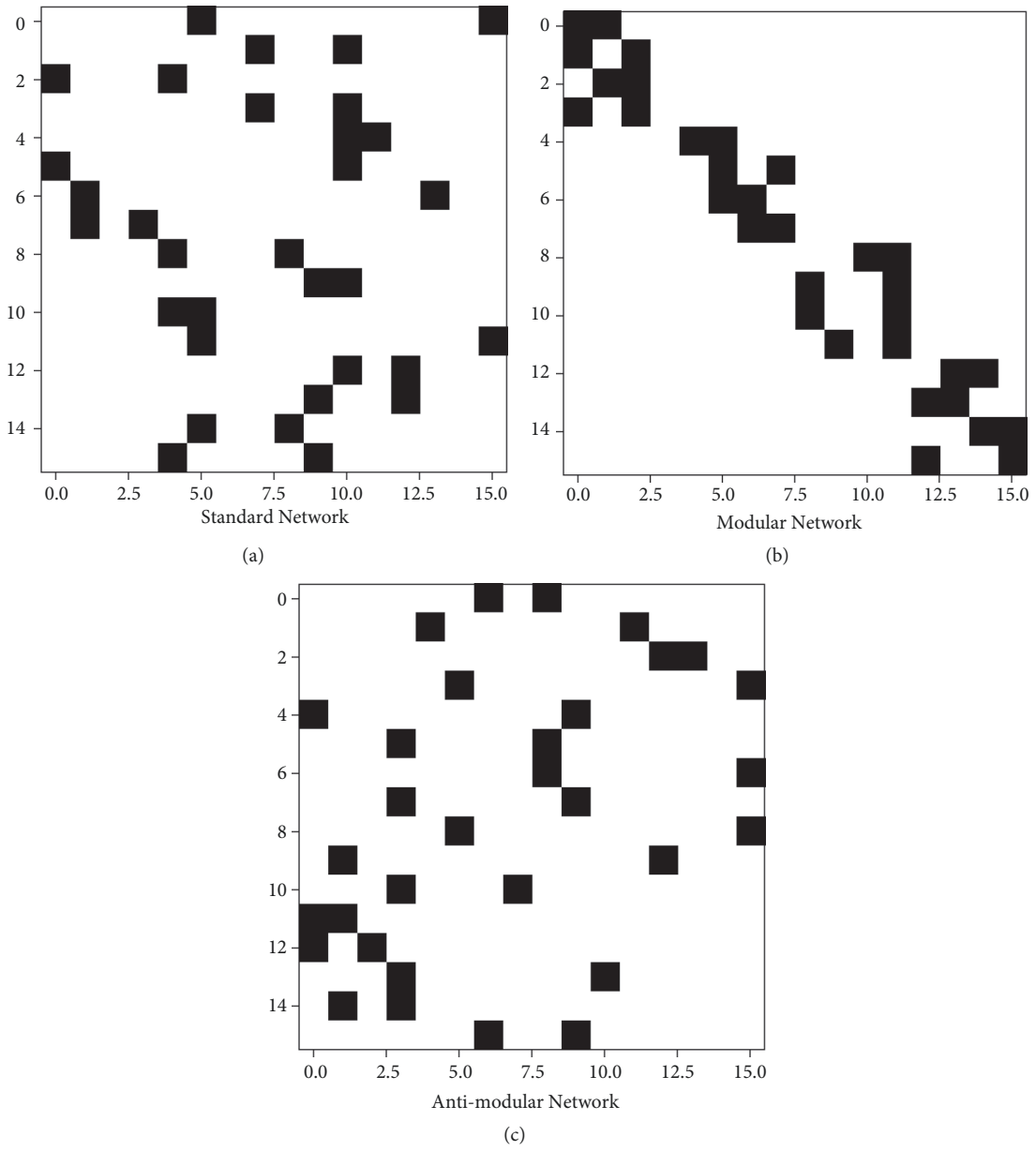


Figure 17: Adjacency matrixes of standard, modular, and antimodular RBN with $N = 16$, $K = 2$.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

We are grateful to Darío Alatorre, Ewan Colman, Omar Karim Pineda, José Luis Mateos, Dante Pérez, and Fernanda Sánchez-Puig for useful comments and discussions. This research was partially supported by CONACYT and DGAPA, UNAM.

References

[1] J. L. McClelland and D. E. Rumelhart, "Distributed memory and the representation of general and specific information," *Journal of Experimental Psychology: General*, vol. 114, no. 2, pp. 159–188, 1985.

[2] J. L. McClelland and D. E. Rumelhart, "A distributed model of human learning and memory," *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, vol. 2, pp. 170–215, 1986.

[3] R. Poblanno-Balp and C. Gershenson, "Modular random Boolean networks," *Artificial Life*, vol. 17, no. 4, pp. 331–351, 2011.

[4] G. Schlosser and G. P. Wagner, *Modularity in Development and Evolution*, The University of Chicago Press, 2004.

[5] E. Segal, M. Shapira, A. Regev et al., "Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data," *Nature Genetics*, vol. 34, no. 2, pp. 166–176, 2003.

[6] A. Hintze and C. Adami, "Modularity and anti-modularity in networks with arbitrary degree distribution," *Biology Direct*, vol.

5, no. 1, article no. 32, 2010.

[7] C. Werner, D. Rasskin-Gutman, and H. A. Simon, *Modularity: Understanding the Development and Evolution of Natural Complex Systems*, MIT Press, 2005.

[8] C. Emmeche, S. Køppe, and F. Stjernfelt, "Levels, emergence, and three versions of downward causation," in *Downward Causation: Minds, Bodies and Matter*, 2000.

[9] D. T. Campbell, "Downward causation in hierarchically organised biological systems," in *Studies in the Philosophy of Biology*, F. J. Ayala and T. Dobzhansky, Eds., Palgrave, London, UK, 1974.

[10] M. A. Bedau, "Downward causation and the autonomy of weak emergence," *Principia*, vol. 6, no. 1, pp. 5–50, 2002.

[11] C. Gershenson, "Introduction to random Boolean networks," https://arxiv.org/abs/nlin/0408006, 2004.

[12] S. A. Kauffman, "Metabolic stability and epigenesis in randomly constructed genetic nets," *Journal of Theoretical Biology*, vol. 22, no. 3, pp. 437–467, 1969.

[13] S. A. Kauffman, *The Origins of Order Self-Organization and Selection in Evolution*, Oxford University Press, 1993.

[14] D. M. Lorenz, A. Jeng, and M. W. Deem, "The emergence of modularity in biological systems," *Physics of Life Reviews*, vol. 8, no. 2, pp. 129–160, 2011.

[15] C. Y. Baldwin and K. B. Clark, *Design Rules: The Power of Modularity*, The MIT Press, 2000.

[16] S. Fortunato, "Community detection in graphs," *Physics Reports*, vol. 486, no. 3–5, pp. 75–174, 2010.

[17] N. Fernández, C. Maldonado, and C. Gershenson, "Information measures of complexity, emergence, self-organization, homeostasis, and autopoiesis," in *Guided Self-Organization: Inception*, M. Prokopenko, Ed., vol. 9 of *Emergence, Complexity and Computation*, pp. 19–51, Springer, Berlin, Germany, 2014.

[18] G. Santamaría-Bonfil, C. Gershenson, and N. Fernández, "A Package for measuring emergence, self-organization, and complexity based on shannon entropy," *Frontiers in Robotics and AI*, vol. 4, p. 10, 2017.

[19] A. Fauré, A. Naldi, C. Chaouiya, and D. Thieffry, "Dynamical analysis of a generic Boolean model for the control of the mammalian cell cycle," *Bioinformatics*, vol. 22, no. 14, pp. e124–e131, 2006.

[20] M. E. Martinez-Sanchez, L. Mendoza, C. Villarreal, and E. R. Alvarez-Buylla, "A minimal regulatory network of extrinsic and intrinsic factors recovers observed patterns of CD4+ T cell differentiation and plasticity," *PLoS Computational Biology*, vol. 11, no. 6, Article ID e1004324, 2015.

[21] D. P. A. Cohen, L. Martignetti, S. Robine, E. Barillot, A. Zinovyev, and L. Calzone, "Mathematical modelling of molecular pathways enabling tumour cell invasion and migration," *PLoS Computational Biology*, vol. 11, no. 11, Article ID e1004571, 2015.

[22] E. Ortiz-Gutiérrez, K. García-Cruz, E. Azpeitia, A. Castillo, M. de la Paz Sánchez, and E. R. Álvarez-Buylla, "A dynamic gene regulatory network model that recovers the cyclic behavior of Arabidopsis thaliana cell cycle," *PLoS Computational Biology*, vol. 11, no. 9, Article ID e1004486, 2015.

[23] L. Calzone, L. Tournier, S. Fourquet et al., "Mathematical modelling of cell-fate decision in response to death receptor engagement," *PLoS Computational Biology*, vol. 6, no. 3, Article ID e1000702, 15 pages, 2010.

[24] S. N. Steinway, M. B. Biggs, T. P. Loughran, J. A. Papin, and R. Albert, "Inference of network dynamics and metabolic interactions in the gut microbiome," *PLoS Computational Biology*, vol. 11, no. 6, Article ID e1004338, 2015.

[25] F. Herrmann, A. Groß, D. Zhou, H. A. Kestler, and M. Kühl, "A boolean model of the cardiac gene regulatory network determining first and second heart field identity," *PLoS ONE*, vol. 7, no. 10, Article ID e46798, 2012.

[26] B. Li, J. Lou, Y. Liu, and Z. Wang, "Robust invariant set analysis of boolean networks," *Complexity*, vol. 2019, Article ID 2731395, 8 pages, 2019.

[27] S. Zhu, J. Lou, Y. Liu, Y. Li, and Z. Wang, "Event-triggered control for the stabilization of probabilistic boolean control networks," *Complexity*, vol. 2018, Article ID 9259348, 9 pages, 2018.

[28] L. Tong, Y. Liu, Y. Li, J. Lu, Z. Wang, and F. E. Alsaadi, "Robust control invariance of probabilistic boolean control networks via event-triggered control," *IEEE Access*, vol. 6, pp. 37767–37774, 2018.

Hindawi
Complexity
Volume 2019, Article ID 3164692, 11 pages
https://doi.org/10.1155/2019/3164692

Research Article

Solutions to All-Colors Problem on Graph Cellular Automata

Xiaoyan Zhang [1] **and Chao Wang** [2]

[1] *School of Mathematical Science & Institute of Mathematics, Nanjing Normal University, Jiangsu 210023, China*
[2] *College of Software, Nankai University, Tianjin 300350, China*

Correspondence should be addressed to Chao Wang; wangchao@nankai.edu.cn

Received 27 December 2018; Revised 20 March 2019; Accepted 8 May 2019; Published 2 June 2019

Academic Editor: Jose C. Valverde

The All-Ones Problem comes from the theory of σ^+-automata, which is related to graph dynamical systems as well as the Odd Set Problem in linear decoding. In this paper, we further study and compute the solutions to the "All-Colors Problem," a generalization of "All-Ones Problem," on some interesting classes of graphs which can be divided into two subproblems: Strong-All-Colors Problem and Weak-All-Colors Problem, respectively. We also introduce a new kind of All-Colors Problem, k-Random Weak-All-Colors Problem, which is relevant to both combinatorial number theory and cellular automata theory.

1. Introduction

A graph dynamical system (GDS) is a dynamical system constructed over a graph whose vertices can have different states, such that all these states together at a given time constitute a state of the system which can evolve according to an updating scheme [1, 2]. The states of the vertices are commonly modeled by the Boolean values 0 and 1, while the updating scheme consists of as many local functions as vertices and a series of rules that indicate the order in which the local functions act. GDS can be divided into two categories: parallel (PDS) [3, 4] and sequential (SDS) [5, 6] when all the local functions act synchronously or follow an order to act, respectively. In the specific literature, other related topics appeared previously, such as Boolean networks (BN) [7] and cellular automata (CA) [8], which are, in fact, particular cases of GDS. In this paper we will consider "All-Colors Problem," which is concerned with graph cellular automata as one kind of dynamical systems on networks.

A graph is a pair $G = (V, E)$ where $E \subseteq V \times V$. These structures allow for self loops. In the theory of automata, we will insist that G is local finite; i.e., every vertex in G is adjacent to only finite many vertices. In this paper, we only consider G as a finite simple graph. It is convenient to identify E with the adjacency matrix of G constructed as a matrix in $\prod_{V\times V} F_2$. Here $F_2 = \{0, 1\}$ is the two-element field and there is a 1 in the u^{th} row and v^{th} column of E if and only if there is a directed edge from vertex u to vertex v in G.

A vertex u is a predecessor of v if there exists an edge (u, v) in G. The collection of all predecessors of v will be denoted by

$$\Gamma_G(v) = \{u \in V \mid (u, v) \in E\}. \tag{1}$$

Note that $\Gamma_G(v)$ may or may not include v depending on whether v has a self-loop in G or not. We will refer to $\Gamma_G(v)$ as the *neighborhood* of v in G. A *configuration* of G is a function

$$X : V \longrightarrow F_2. \tag{2}$$

The collection of all configurations of G will be denoted by C_G. Define the *transition rule* $\sigma_G : C_G \longrightarrow C_G$ by

$$\sigma(X)(v) \coloneqq \sum_{u\in\Gamma(v)} X(u) \mod 2. \tag{3}$$

$A = (G, \sigma_G)$ is called the σ-*automaton* on G.

Configurations are conveniently identified with subsets of V; i.e., X is identified with $\{v \in V \mid X(v) = 1\}$. Observe that algebraically C_G is a vector space over F_2, $C_G = \prod_V F_2$; addition here amounts to take symmetric differences. We will call this space the *configuration space*. Furthermore, σ is a linear map from the configuration space to itself (such rules are called additive in [9]). If one thinks of configuration X as

a column vector over F_2, it is obvious from the definition that $\sigma(X) = E \cdot X$ where E is the adjacency matrix of G.

A σ-automaton $A = (G, \sigma_G)$ is *symmetric* if and only if the adjacency matrix E of G is symmetric; thus symmetric σ-automata arise from undirected graphs. In the following sections, we will consider only symmetric σ-automata on undirected graphs.

Let G be an undirected graph without self-loops and D a subset of V. Define $G(D)$ to be the graph obtained from G by adding self-loops at all vertices in D. σ-automata of the form $(G, \sigma_{G(V)})$ or $(G, \sigma_{G(\emptyset)})$ are called *Lindenmayer automata* on G.

To lighten notation we will usually omit the subscript G and write $\Gamma(v)$, σ and (G, σ), and so forth. Also we will write σ^+ for $\sigma_{G(V)}$ and σ^- for $\sigma_{G(\emptyset)}$. We will write $\mathbf{0}$ for the empty set and $\mathbf{1}$ for V as members of C_G, so $\mathbf{0}(v) = 0$ and $\mathbf{1}(v) = 1$ for all v in V.

The term *All-Ones Problem* was introduced by Sutner; see [10]. It has applications in linear cellular automata; see [11] and the references therein. The problem is cited as follows: suppose each of the vertices of an undirect graph with n vertices is equipped with an indicator light and a button. If the button of a vertex is pressed, the light of that vertex will change from off to on and vice versa; the same happens to the lights of all the edge-adjacent vertices. Initially all lights are off. Now, consider the following questions: is it possible to press a sequence of buttons in such a way that in the end all lights are on? This is referred as the *All-Ones Problem*. If there is such a way, how to find a such way? Here and in what follows, we consider connected simple undirected graphs only. When disconnected graphs must be concerned, we can deal with them component by component. For all terminology and notations on graphs, we refer to [12]. An equivalent version of the All-Ones Problem was proposed by Peled in [13], where it was called the *Lamp Lighting Problem*. We cite it as follows.

Light bulbs $L_1, L_2, \ldots, L_n$ are controlled by switches $S_1, S_2, \ldots, S_n$. Switch S_i changes the on/off status of light L_i and possibly the status of some other lights. Assume that if S_i changes the status of light L_j, then S_j changes the status of light L_i. Initially all the lights are off. Prove that it is possible to operate the switches in such a way that all the lights are on.

The rule of the All-Ones Problem is σ^+-rule on graphs, which means that a button lights not only its neighbors but also its own light L_i. If a button lights only its neighbors but not its own light, this rule on graphs is called σ^--rule. There were many publications on the All-Ones Problem; see Sutner [14, 15], Barua et al. [16], and the references therein. Using linear algebra, Sutner [17] proved that it is always possible to light every lamp in any graphs by σ^+ rule. Lossers [18] gave another beautiful proof also by using linear algebra. A graph-theoretic proof was given by Erikisson et al. [19]. More results and related references can be referred to [20–25].

In graph-theoretic terminology, a solution to the All-Ones Problem with σ^+-rule can be stated as follows: given a graph $G = (V, E)$, where V and E denote the vertex-set and the edge-set of G, respectively, a subset X of V is a solution if and only if for every vertex v of G the number of vertices in X adjacent to or equal to v is odd. Such a subset X is called an *odd parity cover* in [17]. So, the All-Ones Problem can be formulated as follows: given a graph $G = (V, E)$, does a subset X of V exist such that, for any vertex $v \in V - X$, the number of vertices in X adjacent to v is odd, while for any vertex $v \in X$, the number of vertices in X adjacent to v is even?

Sutner [10] proposed the question whether there is a graph-theoretic method to find a solution for the All-Ones Problem for trees. Galvin [26] solved this question.

The "All-Ones Problem" can be considered to find the preimages of configuration $\mathbf{1}$ under parallel map defined by local rule σ. In this paper, we further study the "All-Colors Problem," a natural generalization of "All-Ones Problem," which can be divided into two subproblems: Strong-All-Colors Problem and Weak-All-Colors Problem, respectively.

This paper is organized as follows. In Section 2, we introduce the preliminary and definitions of the "All-Colors Problem" under σ^+-rule. The "All-Colors Problem" can be divided into two subproblems: Strong-All-Colors Problem and Weak-All-Colors Problem which are studied in Sections 3 and 4, respectively. In the end of Section 4, we also introduce a new kind of All-Color Problem, k-Random Weak-All-Colors Problem, which is relevant to both combinatorial number theory and cellular automata theory.

2. Preliminary of All-Colors Problem

Here we introduce a natural generalization for the All-Ones Problem—*All-Colors Problem*. When we discuss the All-Colors Problem, we need a positive integer $m > 2$ and a graph G each of whose vertices has a color value between 0 and $m - 1$. There are two kinds of All-Colors Problem: Strong-All-Colors Problem and Weak-All-Colors Problem. We give their accurate definitions as follows.

Definition 1 (strong-all-colors problem under σ^+-rule on Z_m). If a vertex is pressed one time, then the color values of the vertex and its neighbors are added by 1 under the meaning of modular m. If the initial status is that the color value of every vertex is 0, then how to press some vertices (maybe many times) to make the color value of every vertex equal to 1 under the meaning of modular m.

Definition 2 (weak-all-colors problem under σ^+-rule on Z_m). If a vertex is pressed one time, then the color values of the vertex and its neighbors are added by 1 under the meaning of modular m. If the initial status is that the color value of every vertex is 0, then how to press some vertices (maybe many times) to make the color value of every vertex not equal to 0 under the meaning of modular m.

A solution to the Strong-All-Colors Problem on a graph G under σ^+-rule is a configuration $f : V(G) \longrightarrow Z_m$ such that, for any $v \in V(G)$,

$$\sum_{w \in N(v) \cup \{v\}} f(w) = 1 \mod m, \tag{4}$$

where $N(v)$ is denoted as the set of vertices of G which are adjacent to w. Correspondingly, a solution to the Weak-All-Colors Problem on a graph G under σ^+-rule is a configuration $f : V(G) \longrightarrow Z_m$ such that, for any $v \in V(G)$,

$$\sum_{w \in N(v) \cup \{v\}} f(w) \neq 0 \mod m. \tag{5}$$

It is worth noting that we only need to study the Strong-All-Colors Problem and the Weak-All-Colors Problem in the case that m is a power of a prime. Indeed, let $m = \prod_{i=1}^{r} p_i^{e_i}$ be the prime factorization of m. Suppose that there exists a solution $c_i : V(G) \longrightarrow Z_{p_i^{e_i}}$ to the Strong-All-Colors Problem under σ^+ on $Z_{p_i^{e_i}}$ for any i. That is, for any $v \in V(G)$,

$$\sum_{w \in N_G(v) \cup \{v\}} c_i(w) = 1 \mod p_i^{e_i}. \tag{6}$$

By the Chinese Remainder Theorem, there exists $c : V(G) \longrightarrow Z_m$ such that

$$c(v) = c_i(v) = 1 \mod p_i^{e_i} \tag{7}$$

for any $i \in \{1, \ldots, r\}$ and $v \in V(G)$. It follows that

$$\sum_{w \in N_G(v) \cup \{v\}} c(w) = 1 \mod p_i^{e_i}, \tag{8}$$

for any v. By the Chinese Remainder Theorem again, one has

$$\sum_{w \in N_G(v) \cup \{v\}} c(w) = 1 \mod m. \tag{9}$$

That is, c is a solution to the Strong-All-Colors Problem on Z_m. The discussion of the Weak-All-Colors Problem is similar.

Note that we can also define the corresponding *Strong-All-Colors Problem* under σ^--rule on Z_m and *Weak-All-Colors Problem* under σ^--rule on Z_m; however, they are more difficult to study.

In the following sections, we will discuss the All-Colors Problem under σ^+-rule in detail. We need to use two definitions equivalent to Definitions 1 and 2 for convenience. Suppose G is a simple undirected graph; each vertex of G has a color value on Z_m which can change with time. These changes abide by the following rules (σ^+-rule): If, for $\forall v \in G$, $c_0(v)$ is the initial color value of v at time $t = 0$, then, at time $t = 1$, the color value of v has changed to $c_1(v) = \sum_{u \in N(v) \cup \{v\}} c_0(u)$, where $N(v)$ is the set of vertices which are adjacent to v.

The Strong-All-Colors Problem under σ^+-rule can be redefined as follows: *to find the initial color values of all the vertices of G at time $t = 0$ such that any vertex of G has a color value equal to $1 (\mathrm{mod}\, m)$ at time $t = 1$ under σ^+-rule.*

The Weak-All-Colors Problem can be redefined as follows: *to find the initial color values of all the vertices of G at time $t = 0$ such that no vertex of G has a color value equal to $0 (\mathrm{mod}\, m)$ at time $t = 1$ under σ^+-rule.*

For the sake of simplicity, the Strong-All-Colors Problem is denoted as SACP from now on. Correspondingly, the Weak-All-Colors Problem is denoted as WACP.

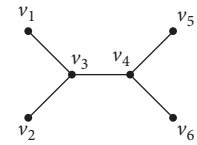


Figure 1: A tree without solution to the SACP for $m \geq 3$.

3. Strong-All-Colors Problem under σ^+-Rule on Z_m

First of all, for a general graph, the SACP with σ^+-rule on Z_m may have no solution. For example, there is no solution to the SACP under σ^+-rule on Z_m for a circle C_n when $3 \nmid n$ and $3 \mid m$. Particularly, by observing the following tree T_0 with 6 vertices, in Figure 1, we will discover surprisingly that there is no solution to the SACP under σ^+-rule on Z_m for any $m \geq 3$.

Then a question arises naturally: How to determine whether a graph has a solution to the SACP with σ^+-rule on Z_m?

It is easy to find an algebraic method to solve this problem. Suppose G is a graph with n vertices. The following theorem can be obtained easily and we omit the proof.

Theorem 3. *There exists a solution to the SACP with σ^+-rule on $Z_m (m \geq 2)$ for a graph G if and only if the following system of linear equations has a solution on Z_m.*

$$(A + I) X = (\mathbf{1}) (\mathrm{mod}\, m), \tag{10}$$

where A is the adjacent matrix of G, I is the identity matrix, X is an $n \times 1$ vector of variables, and $(\mathbf{1})(\mathrm{mod}\, m)$ is an $n \times 1$ vector each element of which is $1 (\mathrm{mod}\, m)$.

Example 4. Consider the tree T_0 at the beginning of this section. Its adjacent matrix is

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}. \tag{11}$$

It is easy to verify that

$$(A + I) \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} 1 \mod m \\ 1 \mod m \\ \vdots \\ 1 \mod m \end{pmatrix} \tag{12}$$

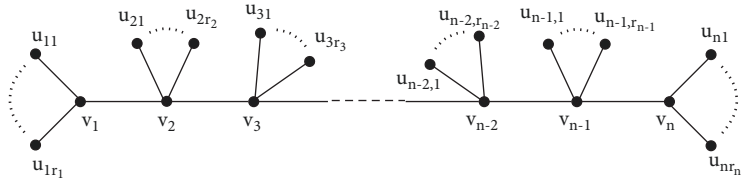


Figure 2: A caterpillar T.

is equivalent to

$$\begin{aligned}
x_1 \quad & \quad +x_3 \quad & & & & & &= 1 \mod m \quad (1)\\
& x_2 \quad +x_3 \quad & & & & & &= 1 \mod m \quad (2)\\
x_1 \; & +x_2 \; +x_3 \; & +x_4 \quad & & & & &= 1 \mod m \quad (3)\\
& & x_3 \quad & +x_4 \; & +x_5 \; & +x_6 \; &&= 1 \mod m \quad (4)\\
& & & x_4 \quad & +x_5 \quad & & &= 1 \mod m \quad (5)\\
& & & x_4 \quad & & +x_6 \; &&= 1 \mod m \quad (6)
\end{aligned} \tag{13}$$

Let us operate the 6 equations as follows.

$$(1) + (2) + (5) + (6) - (3) - (4), \tag{14}$$

so we get

$$0 = 2 \mod m \tag{15}$$

It is obvious that equation (15) holds if and only if $m = 2$. So T_0 has no solution to the SACP under σ^+-rule on Z_m for $m \geq 3$, which we have just declared.

The system of linear equations over Z_m could be solved. Let $A \in R^{n\times n}$, $B \in R^{n\times 1}$, and $X = (x_1, \ldots, x_n)^T$. We need to solve the equation $AX = B$. If R, as Z_m above, is a principle idea ring, there exist invertible matrices $U, V \in M_n(R)$ such that $A = UCV$, where

$$C = \begin{pmatrix} d_1 & 0 & \cdots & 0 \\ 0 & d_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & d_n \end{pmatrix}, \tag{16}$$

where $d_i \mid d_{i+1}$ and C is the Smith normal form of A. We can compute U and V if $R = Z_m$. Then $AX = B$ is equivalent to $CVX = U^{-1}B$. We set $Y = VX$ and we can solve $CY = U^{-1}B$ since C is a diagonal matrix. Hence, we obtain $X = V^{-1}Y$.

Although this algebraic method is quite succinct, it is interesting to study the SACP under σ^+ rule on Z_m over some classes of graphs.

In the following, we consider the SACP under σ^+-rule on Z_m for trees. Note that we do all the operations "+" and "−" on the ring Z_m. Consider the special case when the tree is a "caterpillar." A tree T is called a *caterpillar* if and only if the remainder of T is a path after removing all the leaves of T. Figure 2 shows such a caterpillar.

We can find the solution to the SACP under σ^+-rule on Z_m for a caterpillar T according to the following process.

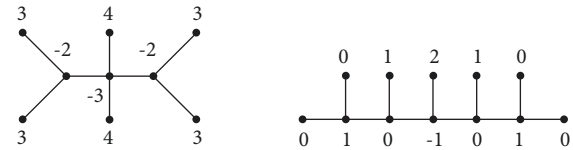


Figure 3: Two examples with solutions to the SACP on Z_m.

Suppose T is a caterpillar with the following shape of Figure 2. First we assume that the initial color value of vertex v_1 is x, which will be determined in the end. It is easy to see that the initial color values of the leaves of vertex v_1 must be $1 - x$ in order to make the color values of vertices $u_{11}, u_{12}, \ldots, u_{1r_1}$ changing to 1 by the meaning of modular m under σ^+-rule. Then the initial color value of v_2 would be $-(r_1 - 1)(1 - x)$ to make the color value of v_1 equal to 1 under σ^+-rule. Repeat this step until we have determined the initial color value of v_n, denoted by $f_n(x)$, which is a linear function of x. From this procedure we can see that there is only one equation to be satisfied; i.e.,

$$f_{n-1}(x) + f_n(x) + r_n \cdot (1 - f_n(x)) = 1 \mod m, \tag{17}$$

where $f_{n-1}(x)$ is the initial color value of vertex v_{n-1}.

So we have the following.

Theorem 5. *A caterpillar T has a solution to the SACP under σ^+-rule on Z_m if and only if the equation (17) has a solution on Z_m.*

We give two examples with solutions to the SACP under σ^+-rule on Z_m in Figure 3. All the negative integers that appeared in the following figures are under the meaning of modular m.

As an application of Theorem 5, we have the following.

Corollary 6. *If T is a path with n vertices, then T has a solution to the SACP under σ^+-rule on Z_m for any $m \geq 2$.*

Proof. Let $x_1 x_2 \cdots x_n$, a sequence of integers on Z_m, represent the initial color values of vertices of T. It is easy to check that

$$x_1 x_2 \cdots x_n = \begin{cases} 100100 \cdots 10010, & \text{if } n \equiv 2 \pmod 3 \\ 100100 \cdots 1001, & \text{if } n \equiv 1 \pmod 3 \\ 010010 \cdots 010, & \text{if } n \equiv 0 \pmod 3 \end{cases} \tag{18}$$

is a solution to the SACP under σ^+-rule on Z_m for any $m \geq 2$. □

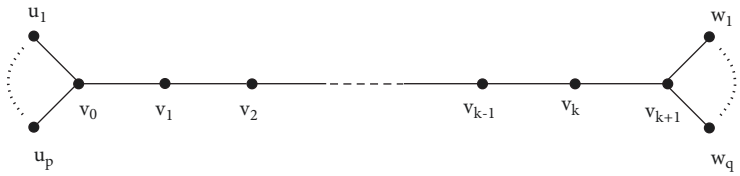


Figure 4: The shape of a special kind of caterpillars.

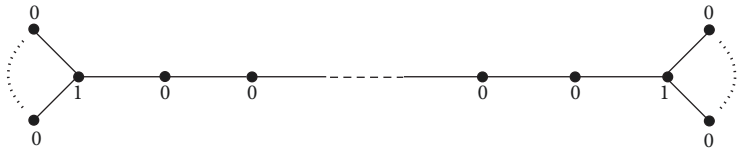


Figure 5: A solution for T when $k \equiv 2 (\mathrm{mod}\, 3)$.

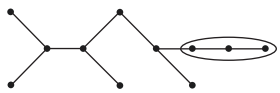


Figure 6: An example of a 3-connected path.

Corollary 7. *If T is a tree with the shape as shown in Figure 4, then*

(1) *when $k \equiv 2 (\mathrm{mod}\, 3)$, T has a solution as in Figure 5*

(2) *when $k \equiv 1 (\mathrm{mod}\, 3)$, T has a solution if and only if $(pq-1)x \equiv q-1 (\mathrm{mod}\, m)$ has a solution on Z_m, which is equivalent to $\gcd(pq-1, m) \mid (q-1)$. Here x is the initial color value of vertices $u_1, \ldots, u_p$*

(3) *when $k \equiv 0 (\mathrm{mod}\, 3)$, T has a solution if and only if $(pq-p-q)x \equiv -q (\mathrm{mod}\, m)$ has a solution on Z_m, which is equivalent to $\gcd(pq-p-q, m) \mid q$. The meaning of x is the same to above*

Now we show a reduction technique. First we introduce a new concept.

Definition 8. Suppose G is a graph. If there are 3 distinct vertices v_1, v_2, v_3 of G satisfying

(1) v_3 is a leaf

(2) v_2 is exactly adjacent to v_1 and v_3

then $(\{v_1, v_2, v_3\}, \{(v_1, v_2), (v_2, v_3)\})$ is called a 3-*connected path* and denoted by (v_1, v_2, v_3) simply.

Example 9. For example, Figure 6 shows a tree which has a 3-connected path marked by an ellipse.

Theorem 10 (deleting a 3-connected path). *Suppose that G is a graph with n vertices $v_1, v_2, \ldots, v_n$. If G contains a 3-connected path $(v_{i_1}, v_{i_2}, v_{i_3})$, then G has a solution to the SACP under σ^+-rule on Z_m if and only if $G \setminus \{v_{i_1}, v_{i_2}, v_{i_3}\}$ has a solution to the SACP under σ^+-rule on Z_m, where $G \setminus \{v_{i_1}, v_{i_2}, v_{i_3}\}$ is a tree or a forest.*

Proof. Suppose T has a solution to the SACP under σ^+-rule on Z_m. Assume the initial color value of vertex v_i is $x_i (1 \le i \le n)$. We have $x_{i_1} = 0 (\mathrm{mod}\, m)$ because it requires $x_2 + x_3 = 1 (\mathrm{mod}\, m)$ and $x_1 + x_2 + x_3 = 1 (\mathrm{mod}\, m)$ to ensure the color value of v_2 and v_3 to be changed to 1 under σ^+-rule. Then we can say that $(x_4, x_5, \ldots, x_n)$ is a solution to the SACP under σ^+-rule on Z_m for $G \setminus \{v_1, v_2, v_3\}$.

For the converse, suppose $(x_4, x_5, \ldots, x_n)$ is a solution to the SACP under σ^+-rule on Z_m for $G \setminus \{v_1, v_2, v_3\}$. Let $\widetilde{N}(v_{i_1})$ denote the set of vertices $G \setminus \{v_1, v_2, v_3\}$ that are adjacent to v_{i_1} in T. It is easy to check $(x_1', x_2', \ldots, x_n')$ is a solution to the SACP under σ^+-rule on Z_m for T, where

$$x_j' = \begin{cases} x_j & 1 \le j \le n \text{ and } j \ne 1, 2, 3 \\ 0 & j = 1 \\ 1 - \sum_{u \in \widetilde{N}(v_{i_1})} x_u & j = 2 \\ \sum_{u \in \widetilde{N}(v_{i_1})} x_u & j = 3 \end{cases} \tag{19}$$

This completes the Proof. □

It is worth noting that we can use this reduction technique repeatedly. For example, the tree in Example 9 can be reduced continuously. The reduction process and its inverse, which can help us to get the solution to the SACP, are shown in Figure 7.

In fact, there are some other complex structures which can be reduced. But they are commonly relevant to Z_m. For example, when $m = 3$, the structures shown in Figure 8 can be removed from a tree, just like the 3-connected path.

Next, we study another special kind of trees—radioactive trees. A tree T is called a *radioactive tree* if and only if there is only at most one vertex of T which has a degree bigger than 2. In fact, a radioactive tree can be viewed as several paths adhering to a common vertex.

Theorem 11. *If T is a radioactive tree, then T has a solution to the SACP under σ^+-rule on Z_m for any $m \ge 2$.*

Proof. Assume that T has the shape shown in Figure 9.

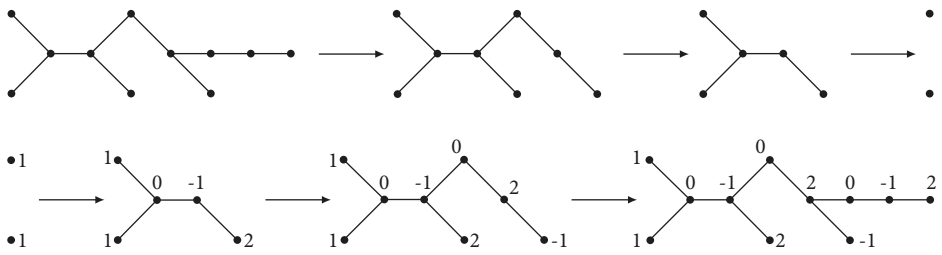


Figure 7: Reduction to get the solution.

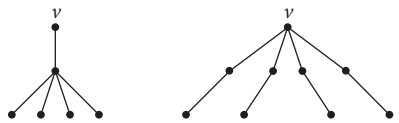


Figure 8: Complex structures that can be reduced when $m = 3$.

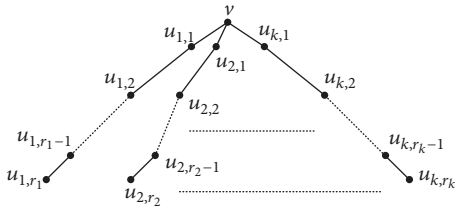


Figure 9: A radioactive tree.

By Theorem 10 we can delete 3-connected path continuously. Without loss of generality we may assume that at last T has changed to T' that has $k' \le k$ paths from the root. It is enough to prove that T' has a solution to the SACP. Suppose $r_i = 1$ for $1 \le i \le s$ and $r_i = 2$ for $s+1 \le i \le k'$ in T'. If $s = k'$, then $x_v = 1$, $x_{u_{i,1}} = 0$ for $1 \le i \le s = k'$ is a solution. If $s < k'$, then $x_v = 0$, $x_{u_{i,1}} = 1$ for $1 \le i \le s$, $x_{u_{i,1}} = 0$ and $x_{u_{i,2}} = 1$ for $s+1 \le i < k'$, $x_{u_{k',1}} = 1 - s(\mathrm{mod}\, m)$ and $x_{u_{k',2}} = s(\mathrm{mod}\, m)$ is a solution to the SACP. The schematic diagram for the latter case is shown in Figure 10. □

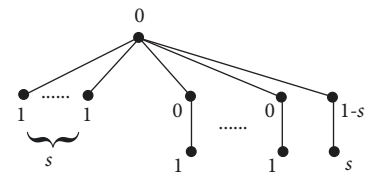


Figure 10: The answer to the SACP for simplified tree T' when $s < k'$.

Actually, we can also prove Theorem 11 with the help of our reduction technique introduced before. The detail of the proof is omitted here.

4. Weak-All-Colors Problem

In this section, we will study the WACP under σ^+-rule on Z_m, $m \ge 2$. It is worth noting when $m = 2$, the WACP becomes the All-Ones Problem which we have discussed before. So the WACP is another generalized form of All-Ones Problem. The WACP has corresponding linear algebraic representation. If G is a graph with n vertices and A is the adjacent matrix of G, then the WACP is equivalent to the next algebraic question:

Find the values of n variables $x_1, x_2, \ldots, x_n$ on Z_m such that no component of the $n \times 1$ result vector of

$$(A + I)\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \tag{20}$$

is equal to 0.

It is known that any graph G has a solution to the All-Ones Problem under σ^+-rule. We conjecture that it also holds for the WACP.

Conjecture 12. *For any graph G, there is a solution to the WACP under σ^+-rule on $Z_m (m \ge 2)$.*

We should point out that the original proof to the existence of a solution to the All-Ones Problem for any graph G cannot do any effects on the WACP on Z_m when $m \ge 3$. So it seems that we need to find another approach to solve this conjecture.

In this section, we emphasize the trees by using the technique of increasing a leaf. By Theorem 14, we show that Conjecture 12 is valid for trees. By Theorem 15, it also holds for cycles. Using the increasing technique and Theorem 15, we prove that Conjecture 12 is valid for unicyclic graphs, which is Corollary 16. Furthermore, by Theorem 17, it is interesting that we can find a solution, which contains only 0 and 1, to the WACP under σ^+-rule on Z_3 for trees. Correspondingly another Conjecture 18 is proposed.

Lemma 13 (increasing a leaf). *If G is a graph and v is a vertex of G whose degree is 1, i.e., v is a leaf, then G has a solution to the WACP under σ^+-rule on $Z_m (m \ge 3)$ if $G \setminus \{v\}$ does too.*

Proof. Assume $G \setminus \{v\}$ has a solution c_0 to the WACP under σ^+-rule on Z_m. Suppose v is only adjacent to w in G. Then let

$$c_0'(v) \in Z_m \setminus \left\{ -\sum_{u \in N(w) \cup \{w\}} c_0(u), -c_0(w) \right\}, \quad c_0'(u) = c_0(u), \text{ if } u \neq v. \tag{21}$$

Because $m \geq 3$, so we could find a $c_0'(v)$ in $Z_m \setminus \{-\sum_{u \in N(w) \cup \{w\}} c_0(u), -c_0(w)\}$. It is easy to check that c_0' is a solution to the WACP under σ^+-rule on Z_m.

The proof is completed. □

From Lemma 13, we can obtain Theorem 14 easily.

Theorem 14. *Let T be a tree and $v \in V(T)$. Then T has a solution c_0 to the WACP under σ^+-rule on $Z_m (m \geq 3)$ satisfying $c_0(v) = 1$.*

Proof. We use the technique of increasing a leaf. Firstly let $T_1 = \{v\}$ be a one-vertex tree, and let $\widehat{c}_1(v) = 1$, which is a solution to the WACP under σ^+-rule on Z_m satisfying $\widehat{c}_1(v) = 1$. Secondly we can add another vertex u, which is adjacent to v, and the edge (u, v) into T_1. Let T_2 denote the new tree. It is easy to see that there is a solution $\widehat{c}_2$ to the WACP under σ^+-rule on Z_m satisfying $\widehat{c}_2(v) = 1$. Repeat these steps by adding an appropriate vertex each time, until at last we get a solution $\widehat{c}_n$ to the WACP under σ^+-rule on Z_m for T satisfying $\widehat{c}_n(v) = 1$, which is just c_0 that we want. We conclude the proof. □

Since it is not hard to construct a solution, we can obtain the following theorem straightforwardly.

Theorem 15. *There is a solution to the WACP under σ^+-rule on $Z_m (m \geq 3)$ for any cycle.*

In fact we can extend the result to "unicyclic graphs." Recall that a graph G is called unicyclic if it contains a unique cycle. In other words, we can regard a unicyclic graph as a cycle attached with each vertex a rooted tree.

Corollary 16. *If G is a unicyclic graph, then there is a solution to the WACP under σ^+-rule on $Z_m (m \geq 3)$ for G.*

Proof. From Theorem 15, a cycle has a solution to the WACP under σ^+-rule on Z_m. Using the increasing technique repeatedly, we can prove that this statement holds. □

Next, we prove that there is a special solution to the WACP on Z_3 for any tree such that all the color values of vertices are 0 and 1.

Theorem 17. *If T is a tree, then there is a solution c_0 to the WACP under σ^+-rule on Z_3 such that $c_0(v) = 0$ or 1 for any $v \in V(T)$.*

Proof. Suppose T is a rooted tree with root t. We need to introduce three new concepts.

The *small WACP* is to find an initial color value array c on T such that

(1) $c(v) = 0$ or $1, \forall v \in T$

(2) $\sum_{v \in N(u) \cup \{u\}} c(v) \neq 0 (\mathrm{mod}\, 3), \forall u \neq t$

(3) $\sum_{v \in N(t) \cup \{t\}} c(v) = 0$ or $1 (\mathrm{mod}\, 3)$

Then the initial color value array c will be called a solution to the small WACP.

Correspondingly, the *positive WACP* is to find an initial color value array c on T such that

(1) $c(v) = 0$ or $1, \forall v \in T$

(2) $\sum_{v \in N(u) \cup \{u\}} c(v) \neq 0 (\mathrm{mod}\, 3), \forall u \neq t$

(3) $\sum_{v \in N(t) \cup \{t\}} c(v) = 1$ or $2 (\mathrm{mod}\, 3)$

Then the initial color value array c will be called a solution to the positive WACP.

The *even WACP* is to find an initial color value array c on T such that

(1) $c(v) = 0$ or $1, \forall v \in T$

(2) $\sum_{v \in N(u) \cup \{u\}} c(v) \neq 0 (\mathrm{mod}\, 3), \forall u \neq t$

(3) $\sum_{v \in N(t) \cup \{t\}} c(v) = 0$ or $2 (\mathrm{mod}\, 3)$

Then the initial color value array c will be called a solution to the even WACP.

In fact a solution to the positive WACP is also a solution we required in this proposition. So it is sufficient for us to prove that there is a solution to the positive WACP for any rooted tree.

Suppose T is a rooted tree with root t. If there is a solution c to the small WACP such that $c(t) = 0$, we say that 0 is a permissible color value of root t to the small WACP. Accordingly, if there is a solution c to the small WACP such that $c(t) = 1$, then 1 is a permissible color value of root t to the small WACP. We name the set of all permissible color values of root t as the *permission set of root t to the small WACP* and denote it by S.

Similarly, we can define *the permission set of root t to the positive WACP*, denoted by P, and *the permission set of root t to the even WACP*, denoted by E. Then for the rooted tree T with root t, (S, P, E) will be called the *type* of root t and corresponding rooted tree T. For example, the type of a rooted tree shown in Figure 11 is $(\{1\}, \{0, 1\}, \{0, 1\})$, or denoted by $(1, 01, 01)$ for simplicity.

Next we need to define "*Add Into Operation*." Suppose T_1 and T_2 are two rooted trees with roots t_1 and t_2, respectively. We will get a new rooted tree with root t_2 by adding T_1 into T_2 with t_1 as a child of t_2. This operation is called an *Add Into Operation on T_1 to T_2*. If the result tree is T_3, then we denote the operation as $T_1 \gg T_2 \longrightarrow T_3$. It is not difficult to show that each rooted tree can be derived by doing an Add Into Operation on one smaller rooted tree to another smaller rooted tree.

Furthermore, if the color value array $c^1_{(0,S)}$ on T_1 is a solution to the small WACP for T_1 and $c^2_{(1,S)}$ on T_2 is a solution to the small WACP for T_2, then the color value array $c^3 = c^1_{(0,S)} \cup c^2_{(1,S)}$ will be a solution to the small WACP for $T_1 \gg T_2$,

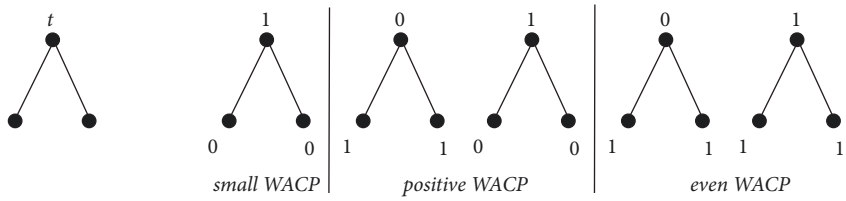


Figure 11: A little tree with root t and its solutions to the three kinds of WACP, which show that the type of it is $(1, 01, 01)$.

i.e., a new tree with root t_2, so we write c^3 as $c^3_{(1,S)}$. Then we denote it as the equation

$$c^1_{(0,S)} \geqslant c^2_{(1,S)} \longrightarrow c^3_{(1,S)}. \tag{22}$$

Similarly, we can get other equations like this. Altogether there are 15 equations as follows.

$$\begin{aligned}
&c^1_{(0,S)} \geqslant c^2_{(1,S)} \longrightarrow c^3_{(1,S)},\\
&c^1_{(0,S)} \geqslant c^2_{(1,P)} \longrightarrow c^3_{(1,P)},\\
&c^1_{(0,S)} \geqslant c^2_{(1,E)} \longrightarrow c^3_{(1,E)},\\
&c^1_{(1,S)} \geqslant c^2_{(1,S)} \longrightarrow c^3_{(1,P)},\\
&c^1_{(1,S)} \geqslant c^2_{(1,P)} \longrightarrow c^3_{(1,E)},\\
&c^1_{(1,S)} \geqslant c^2_{(1,E)} \longrightarrow c^3_{(1,S)},\\
&c^1_{(0,P)} \geqslant c^2_{(0,S)} \longrightarrow c^3_{(0,S)},\\
&c^1_{(0,P)} \geqslant c^2_{(0,P)} \longrightarrow c^3_{(0,P)},\\
&c^1_{(0,P)} \geqslant c^2_{(0,E)} \longrightarrow c^3_{(0,E)},\\
&c^1_{(1,P)} \geqslant c^2_{(0,S)} \longrightarrow c^3_{(0,P)},\\
&c^1_{(1,P)} \geqslant c^2_{(0,P)} \longrightarrow c^3_{(0,E)},\\
&c^1_{(1,P)} \geqslant c^2_{(0,E)} \longrightarrow c^3_{(0,S)},\\
&c^1_{(0,E)} \geqslant c^2_{(1,S)} \longrightarrow c^3_{(1,S)},\\
&c^1_{(0,E)} \geqslant c^2_{(1,P)} \longrightarrow c^3_{(1,P)},\\
&c^1_{(0,E)} \geqslant c^2_{(1,E)} \longrightarrow c^3_{(1,E)}.
\end{aligned} \tag{23}$$

From the above equations, we can calculate the type of the rooted tree T_3 obtained from the Add Into Operation on a rooted tree T_1 to another rooted tree T_2, as long as we have known the types of T_1 and T_2. For example, if the root of T_1 has the type $(01, 1, 0)$ and the root of T_2 has the type $(01, 01, 1)$, then the type of the new result rooted tree derived by doing an Add Into Operation on T_1 to T_2 will be $(1, 01, 01)$. The validity lies in the following 5 equations.

$$\begin{aligned}
&c^1_{(0,S)} \geqslant c^2_{(1,S)} \longrightarrow c^3_{(1,S)},\\
&c^1_{(1,P)} \geqslant c^2_{(0,S)} \longrightarrow c^3_{(0,P)},\\
&c^1_{(0,E)} \geqslant c^2_{(1,P)} \longrightarrow c^3_{(1,P)},\\
&c^1_{(1,P)} \geqslant c^2_{(0,P)} \longrightarrow c^3_{(0,E)},\\
&c^1_{(0,E)} \geqslant c^2_{(1,E)} \longrightarrow c^3_{(1,E)}.
\end{aligned} \tag{24}$$

Then we want to list all possible types of rooted trees. In fact, there are only 8 distinct types of rooted trees. Each type and one of its representative rooted trees are shown in Figure 12.

In order to show that there are only 8 distinct types, we need to show that the following assertion is correct.

Assertion. If each of the two rooted trees T_1, T_2 with roots t_1 and t_2, respectively, belongs to one of the 8 types, then the rooted tree T derived by doing an Add Into Operation on T_1 to T_2 belongs to one of the 8 types.

We have mentioned before how to calculate the type of the result rooted tree obtained by doing the Add Into Operation on a rooted tree to another if their types are known. Since all rooted trees can be derived by doing an Add Into Operation on a smaller rooted tree to another smaller one, we can observe that all the types of rooted trees are contained in the set of 8 distinct types above. The proof of the Assertion exists in Table 1. Since the original table is too wide, we divide it into two tables as in Tables 1(a) and 1(b).

We have showed that any rooted tree belongs to one of the 8 types and, in each of the 8 types, the permission set to the positive WACP is not empty. So any rooted tree would have a solution c to the positive WACP, which finishes the proof. □

From Theorem 17, we will think whether there is a solution to the WACP when $m \geq 4$, which is the following conjecture.

Conjecture 18. *Let T be a tree. Then T has a solution to the WACP under σ^+-rule on $Z_m (m \geq 4)$ such that $c_0(v) = 0$ or 1, $\forall v \in T$.*

In the following, we will put forward a special WACP under σ^+-rule on complete graphs.

Let K_n be a complete graph with n vertices $v_1, v_2, \ldots, v_n$. Each vertex v_i has an initial color value $c_0(v_i)$ on Z_m at time $t = 0$. Suppose, at time $t = 1$, the color value of each vertex v_i, denoted by $c_1(v_i)$, is equal to the sum of the color values of random m vertices under the meaning of modular m, i.e.,

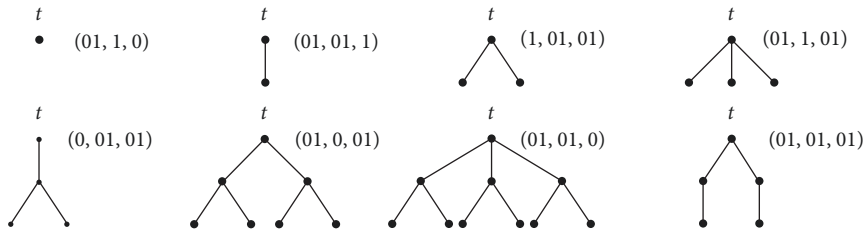


Figure 12: All possible types of rooted trees.

Table 1: The result table of types of a new rooted tree derived by doing an Add Into Operation on a rooted tree T_1 which has the type in the first left column to a rooted tree T_2 which has the type in the first top row.

(a)

	(**01**, **1**, **0**)	(**01**, **01**, **1**)	(**1**, **01**, **01**)	(**01**, **1**, **01**)
(**01**, **1**, **0**)	(01, 01, 1)	(1, 01, 01)	(01, 1, 01)	(01, 01, 1)
(**01**, **01**, **1**)	(01, 01, 01)	(01, 01, 01)	(01, 01, 01)	(01, 01, 01)
(**1**, **01**, **01**)	(0, 01, 01)	(01, 01, 01)	(01, 01, 01)	(01, 01, 01)
(**01**, **1**, **01**)	(01, 01, 1)	(1, 01, 01)	(01, 1, 01)	(01, 01, 1)
(**0**, **01**, **01**)	(01, 01, 0)	(01, 01, 01)	(01, 01, 01)	(01, 01, 01)
(**01**, **0**, **01**)	(01, 1, 01)	(01, 01, 1)	(1, 01, 01)	(01, 1, 01)
(**01**, **01**, **0**)	(01, 01, 01)	(01, 01, 01)	(01, 01, 01)	(01, 01, 01)
(**01**, **01**, **01**)	(01, 01, 01)	(01, 01, 01)	(01, 01, 01)	(01, 01, 01)

(b)

	(**0**, **01**, **01**)	(**01**, **0**, **01**)	(**01**, **01**, **0**)	(**01**, **01**, **01**)
(**01**, **1**, **0**)	(01, 01, 01)	(01, 01, 01)	(01, 01, 01)	(01, 01, 01)
(**01**, **01**, **1**)	(01, 01, 01)	(01, 01, 01)	(01, 01, 01)	(01, 01, 01)
(**1**, **01**, **01**)	(01, 0, 01)	(01, 01, 0)	(0, 01, 01)	(01, 01, 01)
(**01**, **1**, **01**)	(01, 01, 01)	(01, 01, 01)	(01, 01, 01)	(01, 01, 01)
(**0**, **01**, **01**)	(0, 01, 01)	(01, 0, 01)	(01, 01, 0)	(01, 01, 01)
(**01**, **0**, **01**)	(01, 01, 01)	(01, 01, 01)	(01, 01, 01)	(01, 01, 01)
(**01**, **01**, **0**)	(01, 01, 01)	(01, 01, 01)	(01, 01, 01)	(01, 01, 01)
(**01**, **01**, **01**)	(01, 01, 01)	(01, 01, 01)	(01, 01, 01)	(01, 01, 01)

$$c_1(v_i) \in \{c_0(v_{j_1}) + c_0(v_{j_2}) + \cdots + c_0(v_{j_m}) \mod m \mid j_1, j_2, \ldots, j_m \text{ are distinct}\}. \quad (25)$$

If there is no possibility for v_i to have the color value $c_1(v_i)$ equal to 0, i.e.,

$$0 \notin \{c_0(v_{j_1}) + c_0(v_{j_2}) + \cdots + c_0(v_{j_m}) \mod m \mid j_1, j_2, \ldots, j_m \text{ are distinct}\}, \quad (26)$$

then we call v_i 0-*avoidable*. If all vertices are 0-avoidable, then we say that the initial color value array $c_0 = (c_0(v_1), c_0(v_2), \ldots, c_0(v_n))$ is *good*. If c_0 has k distinct elements under the meaning of modular m, we call c_0 *k-good*. The problem to find a k-good initial color value array c_0 is called the *k-Random WACP under σ^+-rule on Z_m*. Note that we always assume that $n \geq m$ and $k \leq m$ when we discuss on the k-Random WACP on Z_m.

We need to point out that if n is big enough, the k-Random WACP may have no solution for any k. For example, it is easy to verify that when $m = 2$, $n = 4$, K_4 has no k-good initial color value array for any $1 \leq k \leq 4$; i.e., the k-Random WACP has no solution. In fact, for any 4 integers x_1, x_2, x_3, x_4 such that $x_i = 0$ or 1, $1 \leq i \leq 4$, the set

$$\{x_1 + x_2, x_1 + x_3, x_1 + x_4, x_2 + x_3, x_2 + x_4, x_3 + x_4\} \quad (27)$$

must have a 2-element subset such that the sum of its elements is equal to 0 under the meaning of modular 2.

In the following we will try to find when the k-Random WACP has a solution and how to construct a solution by the help of combinatorial number theory. We need to represent the k-Random WACP by using the terminology in combinatorial number theory as follows:

Whether there exists $A = (a_1, a_2, \ldots, a_n)$, a sequence of elements of Z_m of length n such that the number of distinct a_i's is equal to k, and the sum of any m elements of A is not equal to 0 *under the meaning of modular m.*

It is necessary to introduce Bialostocki Number $f(m, k)$ from combinatorial number theory. Suppose m, k are positive integers. Denote by $f(m, k)$ the least integer g for which

the following holds: If $A = (a_1, a_2, \ldots, a_g)$ is a sequence of elements of Z_m of length g such that the number of distinct a_i's is equal to k, then there are m indices $i_1, \ldots, i_m$ belonging to $\{1, \ldots, g\}$ such that $a_{i_1} + \ldots + a_{i_m} = 0 (\mathrm{mod}\, m)$. It is easy to see that there is a solution to the k-Random WACP for K_n if and only if $n < f(m, k)$.

How to calculate $f(m, k)$ is still an open problem in combinatorial number theory. Until now only part of Bialostocki Numbers $f(n, k)$ have been derived. In [27], Gallardo, Grekos, and Pihko proved the following.

(1) If m is odd, then $f(m, m) = m$; if m is even, then $f(m, m) = m + 1$.

(2) If $m \geq 5$ and $1 + m/2 < k \leq m - 1$, then $f(m, k) = m + 2$.

In [28], Wang proved the following.

(1) If $k \geq 3$ is odd and $m \geq \max\{(k-1)^2 - 4, (k-1)(k+5)/8 + 2\}$, then

$$f(m, k) = 2n - \frac{(k-1)^2}{4} - 1. \tag{28}$$

(2) If $k \geq 2$ is even and $m \geq \max\{k(k-2) - 4, k(k+2)/8 + 1\}$, then

$$f(m, k) = 2m - \frac{k(k-2)}{4} - 1. \tag{29}$$

Now return to the k-Random WACP. We have the following results.

Theorem 19. *Suppose m and k are two given positive integers and $k \leq m$.*

(1) *When $k = m$ and m is odd, there is no solution to the k-Random WACP on Z_m for K_n.*

(2) *When $k = m$ and m is even, there is a solution to the k-Random WACP on Z_m for K_n if and only if $n = m$, and the initial color value array $c_0 = (0, 1, \ldots, m-1)$ is one solution.*

(3) *When $1 + m/2 < k < m$, there is a solution to the k-Random WACP on Z_m for K_n if and only if $m \leq n \leq m + 1$.*

(4) *When $k \geq 3$ is odd and $m \geq \max\{(k-1)^2 - 4, (k-1)(k+5)/8 + 2\}$, there is a solution to the k-Random WACP on Z_m for K_n if and only if $m \leq n \leq 2m - (k-1)^2/4 - 2$, and the initial color value array*

$$c_0 = \left(\underbrace{0, \ldots, 0}_{x}, \underbrace{1, \ldots, 1}_{y}, 2, 3, \ldots, \frac{k-1}{2}, m - \frac{k-1}{2}, m - \frac{k-1}{2} + 1, \ldots, m-1\right) \tag{30}$$

satisfying $1 \leq x \leq m - (k-1)(k+3)/8$, $1 \leq y \leq m - (k-1)(k+1)/8$, $x + y = n - k + 2$, is one solution.

(5) *When $k \geq 2$ is even and $m \geq \max\{k(k-2) - 4, k(k+2)/8 + 1\}$, there is a solution to the k-Random WACP on Z_m for K_n if and only if $m \leq n \leq 2m - k(k-2)/4 - 2$, and the initial color value array*

$$c_0 = \left(\underbrace{0, \ldots, 0}_{x}, \underbrace{1, \ldots, 1}_{y}, 2, 3, \ldots, \frac{k-1}{2}, m - \frac{k-1}{2}, m - \frac{k-1}{2} + 1, \ldots, m-1\right) \tag{31}$$

satisfying $1 \leq x \leq m - k(k+2)/8$, $1 \leq y \leq m - k(k+2)/8$, $x + y = n - k + 2$, is one solution.

The proof and corresponding solutions constructed can be referred to [27, 28].

5. Conclusion

The All-Ones Problem comes from the theory of σ^+-automata, which is related to the graph dynamical system. In this paper, we introduce and study the generalization form of the All-Ones Problem, named "All-Colors Problem." The "All-Colors Problem" can be divided into Strong-All-Colors Problem and Weak-All-Colors Problem, respectively. We analyze and compute the solutions to these two problems on some interesting classes of graphs. At last we also introduce a new kind of Weak-All-Color Problem, k-Random Weak-All-Colors Problem, which is relevant to both combinatorial number theory and σ^+-automata.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by NSFC (Nos. 11871280 and 11471003) and Qing Lan Project.

References

[1] J. A. Aledo, S. Martinez, and J. C. Valverde, "Graph dynamical systems with general boolean states," *Applied Mathematics & Information Sciences*, vol. 9, no. 4, pp. 1803–1808, 2015.

[2] M. Macauley and H. S. Mortveit, "Cycle equivalence of graph dynamical systems," *Nonlinearity*, vol. 22, no. 2, pp. 421–436, 2009.

[3] J. A. Aledo, S. Martínez, F. L. Pelayo, and J. C. Valverde, "Parallel discrete dynamical systems on maxterm and minterm Boolean functions," *Mathematical and Computer Modelling*, vol. 55, no. 3-4, pp. 666–671, 2012.

[4] J. A. Aledo, S. Martínez, and J. C. Valverde, "Parallel discrete dynamical systems on independent local functions," *Journal of Computational and Applied Mathematics*, vol. 237, no. 1, pp. 335–339, 2013.

[5] J. A. Aledo, S. Martínez, and J. C. Valverde, "Updating method for the computation of orbits in parallel and sequential dynamical systems," *International Journal of Computer Mathematics*, vol. 90, no. 9, pp. 1796–1808, 2013.

[6] C. L. Barrett, W. Y. Chen, and M. J. Zheng, "Discrete dynamical systems on graphs and Boolean functions," *Mathematics and Computers in Simulation*, vol. 66, no. 6, pp. 487–497, 2004.

[7] T. Mihaela, T. Matache, and J. Heidel, "Asynchronous random Boolean network model based on elementary cellular automata rule," *Physical Review E: Statistical, Nonlinear, and Soft Matter Physics*, vol. 71, no. 2, pp. 1–13, 2005.

[8] J. Kari, "Theory of cellular automata: a survey," *Theoretical Computer Science*, vol. 334, no. 1–3, pp. 3–33, 2005.

[9] K. Sutner, "Additive automata on graphsa," *Complex Systems*, vol. 2, no. 1, pp. 1–28, 1988.

[10] K. Sutner, "Problem 88-8," *Mathematical Intelligencer*, vol. 10, 1988.

[11] K. Sutner, "On σ-automata," *Complex Systems*, vol. 2, no. 1, pp. 1–28, 1988.

[12] J. A. Bondy and U. S. R. Murty, *Graph Theory with Application*, Macmillan, London, UK, 1976.

[13] U. Peled, "Problem 10197," *American Mathematical Monthly*, vol. 99, article 162, 1992.

[14] K. Sutner, "The σ-game and cellular automata," *The American Mathematical Monthly*, vol. 97, no. 1, pp. 24–34, 1990.

[15] K. Sutner, "σ-automata and Chebyshev-polynomials," *Theoretical Computer Science*, vol. 230, no. 1-2, pp. 49–73, 2000.

[16] R. Barua and S. Ramakrishnan, "σ-game, σ+-game and two-dimensional additive cellular automata," *Theoretical Computer Science*, vol. 154, no. 2, pp. 349–366, 1996.

[17] K. Sutner, "Linear cellular automata and the Garden-of-Eden," *The Mathematical Intelligencer*, vol. 11, no. 2, pp. 49–53, 1989.

[18] O. P. Lossers, "Solution to problem 10197," *American Mathematical Monthly*, vol. 100, pp. 806-807, 1993.

[19] H. Eriksson, K. Eriksson, and J. Sjöstrand, "Note on the lamp lighting problem," *Advances in Applied Mathematics*, vol. 27, no. 2-3, pp. 357–366, 2001.

[20] B. Curtis, J. Earl, D. Livingston, and B. Shader, "Resolution of conjectures related to Lights out! and cartesian products," *Electronic Journal of Linear Algebra*, vol. 34, pp. 718–724, 2018.

[21] R. Fleischer and J. Yu, "A survey of the game Lights Out!, space-efficient data structures, streams, and algorithms," vol. 8066 of *Part of the Lecture Notes in Computer Science book series*, pp. 176–198, 2013.

[22] H.-w. Huang, "Two-lit trees for lit-only σ-game," *Linear Algebra and its Applications*, vol. 438, no. 3, pp. 1057–1066, 2013.

[23] J. Hughes, "Random lights out processes on graphs," *Advances in Applied Mathematics*, vol. 51, no. 2, pp. 254–265, 2013.

[24] X. Wang and Y. Wu, "Lit-only σ-game on pseudo-trees," *Discrete Applied Mathematics*, vol. 158, no. 17, pp. 1945–1952, 2010.

[25] Y. Wu, "Lit-only sigma game on a line graph," *European Journal of Combinatorics*, vol. 30, no. 1, pp. 84–95, 2009.

[26] F. Galvin, "Solution to problem 88-8," *Mathematical Intelligencer*, vol. 11, pp. 31-32, 1989.

[27] L. Gallardo, G. Grekos, and J. Pihko, "On a variant of the Erdös-Ginzburg-Ziv problem," *Acta Arithmetica*, vol. 89, no. 4, pp. 331–336, 1999.

[28] C. Wang, "Note on a variant of the Erdös-Ginzburg-Ziv problem," *Acta Arithmetica*, vol. 108, no. 1, pp. 53–59, 2003.

Hindawi
Complexity
Volume 2019, Article ID 3728621, 10 pages
https://doi.org/10.1155/2019/3728621

Research Article

A Novel Antifragility Measure Based on Satisfaction and Its Application to Random and Biological Boolean Networks

Omar K. Pineda,[1,2] **Hyobin Kim**,[2] **and Carlos Gershenson**[2,3,4]

[1] *Posgrado en Ciencia e Ingeniería de la Computación, Universidad Nacional Autónoma de México, 04510 CDMX, Mexico*
[2] *Centro de Ciencias de la Complejidad, Universidad Nacional Autónoma de México, 04510 CDMX, Mexico*
[3] *Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas, Universidad Nacional Autónoma de México, 04510 CDMX, Mexico*
[4] *ITMO University, St. Petersburg 199034, Russia*

Correspondence should be addressed to Carlos Gershenson; cgg@unam.mx

Received 17 December 2018; Accepted 23 April 2019; Published 28 May 2019

Academic Editor: Massimiliano Zanin

Antifragility is a property that enhances the capability of a system in response to external perturbations. Although the concept has been applied in many areas, a practical measure of antifragility has not been developed yet. Here we propose a simply calculable measure of antifragility, based on the change of "satisfaction" before and after adding perturbations, and apply it to random Boolean networks (RBNs). Using the measure, we found that ordered RBNs are the most antifragile. Also, we demonstrated that seven biological systems are antifragile. Our measure and results can be used in various applications of Boolean networks (BNs) including creating antifragile engineering systems, identifying the genetic mechanism of antifragile biological systems, and developing new treatment strategies for various diseases.

1. Introduction

Antifragility suggested by Taleb is defined as a property to enhance the capability of a system in response to external stressors [1]. It is beyond resilience or robustness. While the resilient/robust systems resist stress and stay the same, antifragile systems not only withstand stress but also benefit from it. The immune system is a representative example of antifragile systems. When exposed to diverse germs at an early age, our immune system strengthens and thus overcomes new diseases in the future.

The concept of antifragility has been actively applied in numerous areas such as risk analysis [2, 3], physics [4], molecular biology [5, 6], transportation planning [7, 8], engineering [9–11], and aerospace and computer science [12–15]. However, a practical measure of antifragility has not been developed yet. Here we propose a novel measure for antifragility based on the change of complexity. We use random Boolean networks (RBNs) as a case study to illustrate our measure. We quantify the complexity by assessing the extent of how much the node states of a RBN are maintained and changed during state transitions. We perturb the network, flipping the node states with the structure of the network fixed. Calculating the variation of the complexity in the network before and after adding the perturbations, we measure antifragility.

BNs have a wide range of applications from biochemical systems [16–20], to economic systems [21], from social networks, [22, 23] to robots [24]. Our antifragility measure can be utilized in various applications of BNs. For instance, one could create antifragile engineered systems or identify the genetic mechanisms of antifragile biological systems.

The rest of our article is structured as follows. In the section of "Measurement of Antifragility in RBNs", we describe RBNs, complexity of RBNs, perturbations to RBNs, and how to assess antifragility in RBNs. In the section "Experiments", methods and parameter setting for simulations are explained. In the section of "Results and Discussion", the results of the antifragility of RBNs and several biological BNs are presented and analyzed. The section of "Conclusions" summarizes and closes the article.

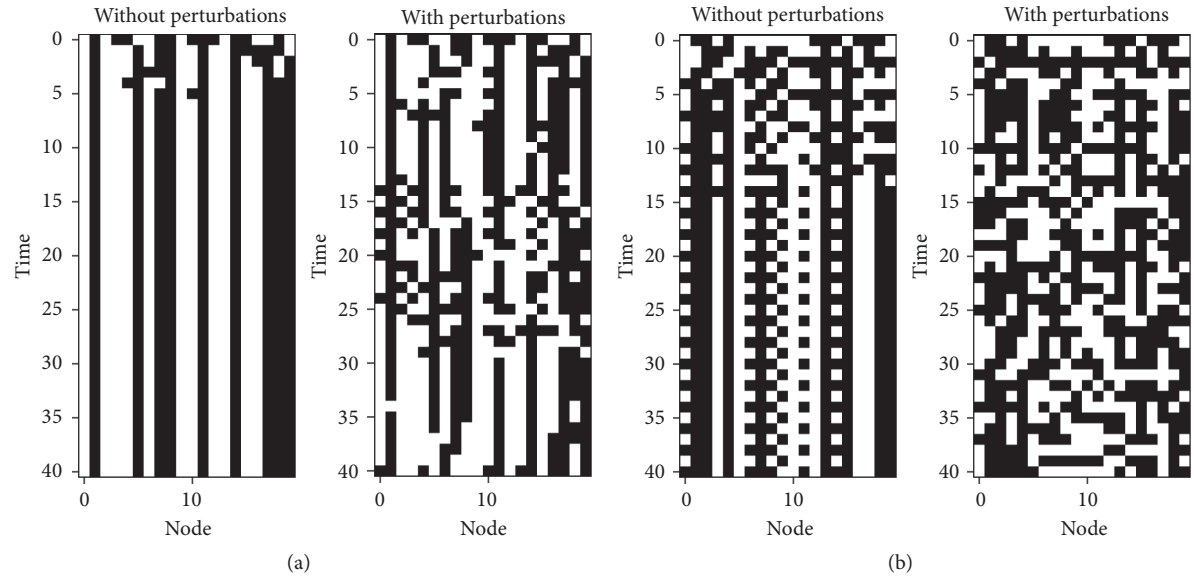


Figure 1: Schematic diagrams showing state transitions of (a) critical and (b) chaotic RBNs with $N = 20$, $X = 2$, and $O = 1$. The left side is the network without perturbations and the right one is the network with perturbations with the same initial states. Each square represents the state of a node (white = 0, black = 1). The state transitions were calculated from the initial states at the top to states at the bottom during $T = 40$. (a) $K = 2$ (critical), $\oint = -0.0958$ (complexity is increased by perturbations: antifragile). (b) $K = 3$ (chaotic), $\oint = 0.0388$ (complexity is decreased by perturbations: fragile).

2. Measurement of Antifragility in RBNs

2.1. Random Boolean Networks. RBNs were proposed as models of gene regulatory networks by Kauffman [26, 27]. A RBN consists of N nodes representing genes. Each node can take either 0 (off, inhibited) or 1 (on, activated) as its state. The node state is determined by the states of input nodes and Boolean functions assigned to each node. Every node has K input nodes (or input links). Self-inputs are allowed. The links are wired randomly, and the Boolean functions are also randomly assigned. Once the links and the Boolean functions set up, they remain fixed.

In Figures 1(a) and 1(b), the left plots show how randomly chosen initial states are updated over time. The plots are simulated until $T = 40$. A state space refers to the set of all the possible configurations (2^N) and all the transitions among them. Being deterministic, classic RBNs have one and only one successor for each state. In the state space, repeated states are attractors, which can be fixed points or limit cycles. The other states that lead to the attractors are basin of attraction of the attractors.

Depending on the structure of the state space, there are three dynamical regimes in RBNs: ordered, chaotic, and critical. The first two are phases, while the critical regime lies at the phase transition. Ordered dynamics are characterized by the change of few node states, which is related to high stability. Chaotic dynamics are characterized by the change of most node states, which is associated with high variability. Critical dynamics balance the stability of the ordered regime with the variability of the chaotic regime [28, 29]. The dynamical regimes can be varied by K. For RBNs with internal homogeneity (i.e., the probability that a gene is activated [30]) $p = 0.5$, $K = 1$ is ordered, $K = 2$ is critical, and $K > 2$ is chaotic, on average [31]. Other properties of RBNs can be used to regulate dynamical regimes [32].

2.2. Complexity of RBNs. It is well known that complex adaptive systems are equipped with stability and flexibility simultaneously. Here complexity signifies a balance between regularity and change, which allows systems to adapt robustly [27, 33, 34]. From an information viewpoint, the regularity ensures that useful information survives, while the change enables the systems to explore new possibilities essential for adaptability. Living organisms or computer systems need not only stability to survive or to maintain information but also flexibility to evolve and adapt to their environment. Following this concept of complexity, we developed a quantitative measure [28]. Using our previous approach, we can measure the complexity of RBNs. In this study, complexity is presented as quantities computed according to our measure.

The complexity is calculated based on Shannon's information entropy. Its equation is as follows:

$$E_i = -\left(p_0 \log_2 p_0 + p_1 \log_2 p_1\right) \quad (1)$$

$$C = 4 \times \overline{E} \times \left(1 - \overline{E}\right) \quad (2)$$

where E_i is the "emergence" of node i, p_j is the probability that the state of the node is j ($j = 0, 1$) among the states of node i updated at each time step until simulation time T, C ($0 \le C \le 1$) is the complexity of the network, and $\overline{E}$ ($0 \le \overline{E} \le 1$) is the average of the emergence values for all the nodes. Specifically, p_0 (p_1) is calculated by counting the number of 0s (1s) in node i until simulation time T. For example, in the left plot of Figure 1(a), p_0 and p_1 of the last node are 2/40 and 38/40, respectively. Because RBNs are deterministic systems, once

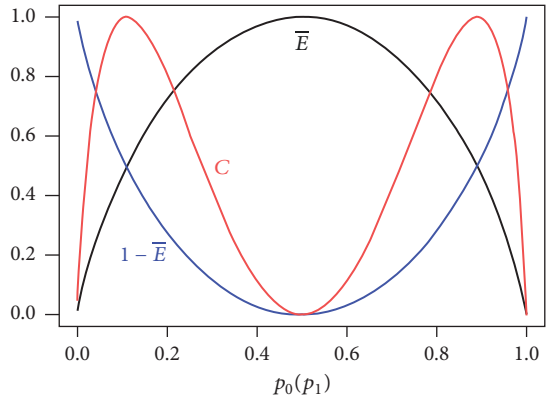


FIGURE 2: Relationship between change $\overline{E}$, regularity $1-\overline{E}$, and complexity C in RBNs [25].

initial states are determined, state transitions from them to attractors are also determined. Thus, $\overline{E}$ and C are dependent on initial states.

When $\overline{E}$ is calculated, p_0 and p_1 in (1) should not be confused with p which was mentioned as internal homogeneity in previous section. p_0 and p_1 are values measured from state transitions. Meanwhile, p is a parameter used to create Boolean functions assigned to each node in a RBN. In the Boolean functions, each value is determined with probability p of being one or probability $1-p$ of being zero.

$\overline{E}$, $1-\overline{E}$, and C are time-dependent because they focus on the dynamics of node states. $\overline{E}$ indicates how much new states are produced over time (i.e., change). As the complement of $\overline{E}$, $1-\overline{E}$ represents how much existing states are maintained (i.e., regularity). C means how successfully both of them are met. Numerically, C reaches maximum when the emergence $\overline{E}$ is 0.5 ($\overline{E}=0.5 \longrightarrow C=1$). It is when the expression of any one of the two states is highly probable, i.e., p_0 or $p_1 \cong 0.89$ for each node [25, 29]. Meanwhile, C becomes 0 when the two states are evenly distributed ($p_0=p_1=0.5$; $\overline{E}=1$) or only one state has maximum probability (p_0 or $p_1=1$; $\overline{E}=0$).

Figure 2 illustrates a mathematical relation between change $\overline{E}$, regularity $1-\overline{E}$, and complexity C in RBNs [25]. As seen in the figure, high complexity is achieved when $\overline{E}=1-\overline{E}$, which means an optimal balance between keeping and changing the states of the network. For perturbed RBNs, Figure 1(a) shows that the antifragile network maintains original states overall and simultaneously explores new states by means of perturbations. Figure 1(b) represents that most of the states in the fragile network change with perturbations, which indicates that the network does not maintain information in a noisy environment.

2.3. Network Perturbations. We express network perturbations due to external stressors as the change of node states in a RBN. We flip the states of X nodes randomly chosen, where the perturbations are added with frequency O during simulation run time T. In other words, the perturbations are added whenever the time step t is divisible by O (t mod $O=0$). For example, $X=2$, $O=3$, and $T=99$ mean that the states of two nodes randomly chosen in each configuration are flipped every three time steps until the simulation run time becomes 99. By comparing the state transitions of the original network and its perturbed network, we can observe how the perturbations propagate over time (Figure 1).

In our study, the degree of perturbations is defined as follows:

$$\Delta x = \frac{X \times (T/O)}{N \times T} \tag{3}$$

where $0 \le \Delta x \le 1$.

2.4. Antifragility of RBNs. We define (*anti*)*fragility* $\oint$ as

$$\oint = -\Delta\sigma \times \Delta x \tag{4}$$

where $\Delta\sigma$ is the difference of "satisfaction" before and after perturbations, while Δx is the degree of perturbations. To prevent the influence of node size of a network, we calibrate the values of $\oint$ by multiplying Δx. The satisfaction σ is the degree to which the goals of an agent have been achieved [35]. In the context of RBNs, each node of the network can be seen as an agent. We can arbitrarily define their goal as reaching a balance between change and regularity, which is achieved when the nodes have high complexity values. Thus, in RBNs, the satisfaction is measured with complexity. Depending on how the satisfaction changes before and after perturbations, the RBN is classified: fragile, robust, or antifragile.

The satisfaction can be measured differently depending on the particular systems, e.g. performance, value, and fitness. If the satisfaction is decreased with perturbations, then the system is fragile. If the satisfaction does not change before and after adding perturbations, then the system is robust. If the satisfaction increases with perturbations, then the system is antifragile. Notice that $\Delta\sigma$ and Δx should be normalized to the interval [-1, 1], [0, 1], respectively.

The perturbations Δx for RBNs were defined in the previous section. We can define the "satisfaction" of a RBN based on its complexity. Since high complexity offers a balance between robustness and adaptability, we can arbitrarily prefer RBNs with high complexity. Using the complexity measure presented previously, $\Delta\sigma$ is calculated by the following equation:

$$\Delta\sigma = C - C_0 \tag{5}$$

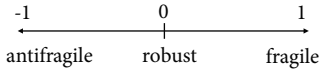
where C_0 is complexity of a network before adding perturbations and C is complexity of the network after adding perturbations. The same initial states are used at $t=0$. Because the value of complexity is between 0 and 1, $-1 \le \Delta\sigma \le 1$.

Negative values of $\oint$ mean that the RBN is antifragile and positive values mean that the RBN is fragile. Values close to zero indicate that the RBN is robust. As shown in (4), $\oint$ has the opposite sign of $\Delta\sigma$. Hence, the negative values of $\oint$ indicate that C is larger than C_0 (i.e., the complexity of a system is improved by external perturbations), while the positive values represent that C_0 is greater than C (i.e., the complexity is lowered by the perturbations). The value of 0

Table 1: Parameter settings for experiments.

Figure	N	T	X	O	# of different networks	# of initial states
3(a)	100	200	1..100	1	50	10
3(b)	100	200	40	1..50	50	10
4(a)	100	2000	0	0	1000	1
4(b)	100	200	1..100	1	50	10
4(c)	100	200	1..100	1	50	10
5	N	200	1..N	1	1	1000
6	100	200	1..100	1..30	50	10
7	N	200	1..N	1..20	1	5000

refers to the fact that complexity does not change before and after perturbations, which indicates that the RBN is robust. Figures 1(a) and 1(b) show the values of $\oint$ calculated from the examples of critical and chaotic RBNs.



In a RBN, the value of $\oint$ can be different depending on initial states because $\Delta\sigma$ is determined by the states of nodes. Thus, using multiple initial states, we calculate average $\oint$ for a RBN and represent it as a system property.

3. Experiments

We performed two sets of experiments: one for RBNs and the other for biological BNs.

First, to measure antifragility of RBNs, we generated ordered, critical, and chaotic RBNs composed of 100 nodes (K = 1 (ordered), 2 (critical), 3, 4, 5 (chaotic)) with internal homogeneity p = 0.5 [31]. For each RBN, we randomly chose 10 different initial states and then examined their state transitions until simulation time T = 200, respectively. For the same RBN taking the same initial states, varying perturbed node size X and perturbation frequency O, we obtained the state transitions of the perturbed RBN until T = 200. By comparing complexity before and after perturbations, we calculated mean of antifragility for the 10 initial states. The measured values shown in the plots are average calculated from 50 different RBNs per K.

Secondly, to measure antifragility of biological BNs, we used the following seven biological network models:

(i) *CD4+ T cell differentiation and plasticity* [36] (N = 18). It is a model representing how *CD4+ T cells* orchestrate immune responses depending on environmental signals and immunological challenges.

(ii) *Mammalian cell-cycle* [37] (N = 20). It is a model explaining the mechanism of action of the cell cycle checkpoints in mammalian cells.

(iii) *Cardiac development* [38] (N = 15). It is a model referring to how the first heart field (FHF) and second heart field (SHF) are formed by differential expression of transcription and signaling factors during cardiac developmental processes.

(iv) *Metabolic interactions in the gut microbiome* [39] (N = 12). It is a model describing interactive host-microbiota metabolic processes.

(v) *Death receptor signaling* [40] (N = 28). It is a model related to the activation of death receptors (TNFR and Fas) that determine either survival or cell death.

(vi) *Arabidopsis thaliana cell-cycle* [41] (N = 14). It is a model explaining the mechanism of plant cell-cycle and cell differentiation in *A. thaliana*.

(vii) *Tumor cell invasion and migration* [42] (N = 32). It is a model representing the mechanism and interplays between pathways that are involved in the process of metastasis.

For each network, we randomly chose 1000 different initial states and then investigated their state transitions until T = 200. Changing X and O, we computed antifragility. Specifications of parameters for the simulation follow Table 1. Our simulator for antifragility was implemented in Python (the source code is available at https://github.com/Okarim1/RBN.git).

4. Results and Discussion

4.1. Antifragility in RBNs. Figure 3 shows average $\oint$ of ordered (K = 1), critical (K = 2), and chaotic (K = 3, 4, 5) RBNs depending on perturbed node size X and perturbation frequency O. The ordered and critical RBNs had negative values (antifragility) in certain ranges of X and O, while the chaotic RBNs all had zero or positive values in the given ranges. This means that the ordered and critical RBNs can be antifragile if they have the "right" amount of perturbations. However, chaotic RBNs are just robust or fragile against perturbations.

As shown in Figure 3(a), the values of the ordered and critical RBNs were lower than zero and got smaller and smaller as X increased, which indicates that their dynamics change more and more to antifragile. However, the values increased beyond certain X values, and even the critical RBNs changed from antifragile to fragile (X > 20). From this, we found that neither too large nor too small, but a moderate level of perturbations can induce greater antifragility. These dynamics are similar to the slower-is-faster effect, where a moderate level of speed can lead to better traffic flow rather than that of the highest speed of individuals [43].

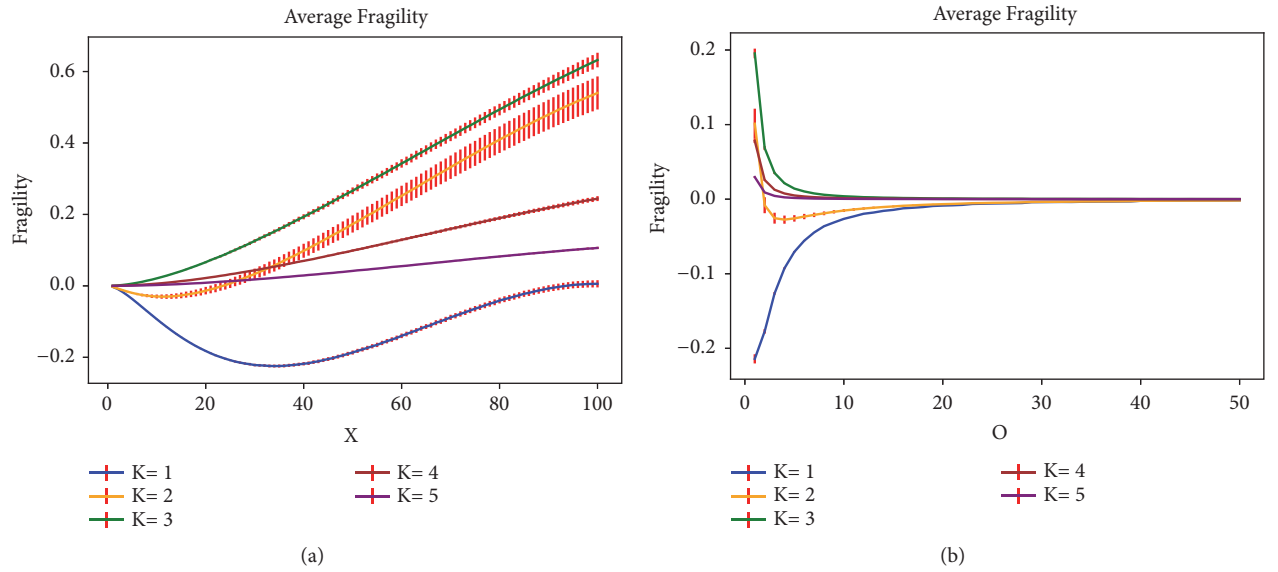


FIGURE 3: Average ϕ of ordered, critical, and chaotic RBNs depending on X and O. The error bars represent the standard error of measurements for 50 different networks at 10 different initial states ran by 200 steps. (a) $N = 100$ and $O = 1$. (b) $N = 100$ and $X = 40$.

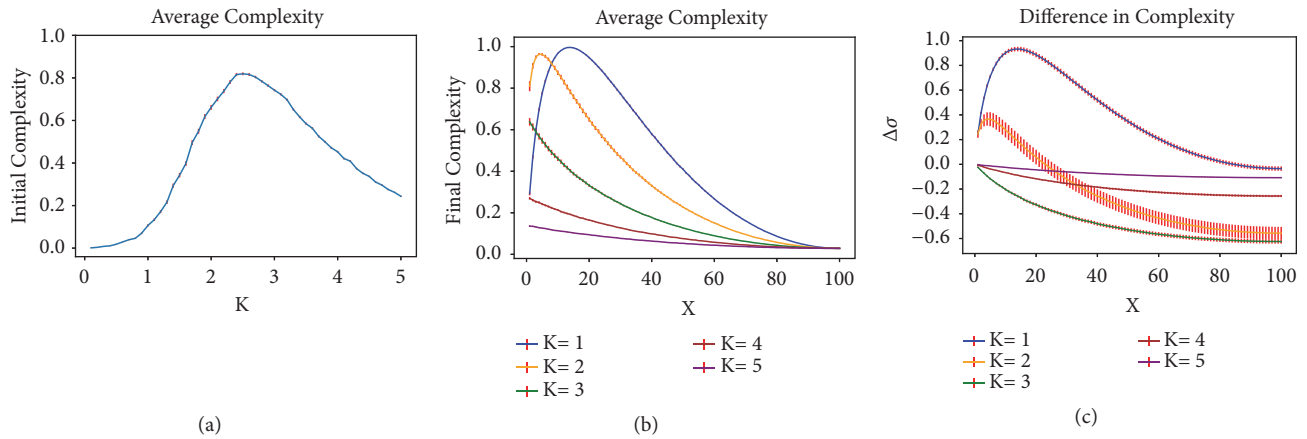


FIGURE 4: Initial and final complexity for $K = 1, 2, 3, 4, 5$ with $N = 100$. The error bars represent the standard error of measurements for 50 different networks at 10 different initial states run by 200 steps. (a) Complexity before adding perturbations. (b) Complexity after adding perturbations. (c) Difference of complexity before and after perturbations.

Meanwhile, in Figure 3(b), antifragility of the ordered and critical RBNs decreased overall as O grew (i.e., the period of adding perturbations became longer and longer). Furthermore, all the RBNs were robust in the case of that the perturbations were not added frequently although the perturbed nodes were 40 ($X = 40$). From these results, we found that the more frequently perturbations are added, the more antifragile a system is, particularly for the ordered RBNs. Moreover, how often perturbations are added has a greater effect on antifragility than how many nodes are perturbed. Thus, it is essential that moderate perturbations are added frequently in order to obtain maximal antifragility.

Based on Figure 3, we are able to see that the ordered RBNs are the most antifragile. Figure 4 clearly accounts for the reason. In Figure 4(a), the complexity before adding perturbations was the lowest at $K = 1$. However, as shown in Figure 4(b), the complexity after adding perturbations increased most greatly and the value was also the largest except for the early range of X at $K = 1$. Therefore, the difference was the largest at $K = 1$ (Figure 4(c)), which led the ordered RBNs to be most antifragile.

Our result for complexity before perturbations is the same as previous studies showing that critical RBNs have the most appropriate balance between regularity and change [25, 28, 44]. In Figure 4(a), for low K, the complexity was low, which represents that the ordered RBNs have high robustness and few changes. That is, there is few or no information emerging. For high K, the complexity was also low, which reflects that the chaotic RBNs have high variability and many changes. Almost all the nodes carry novel emergent information. For medium connectivities ($2 < K < 3$), there was a balance between regularity and change, leading to high complexity.

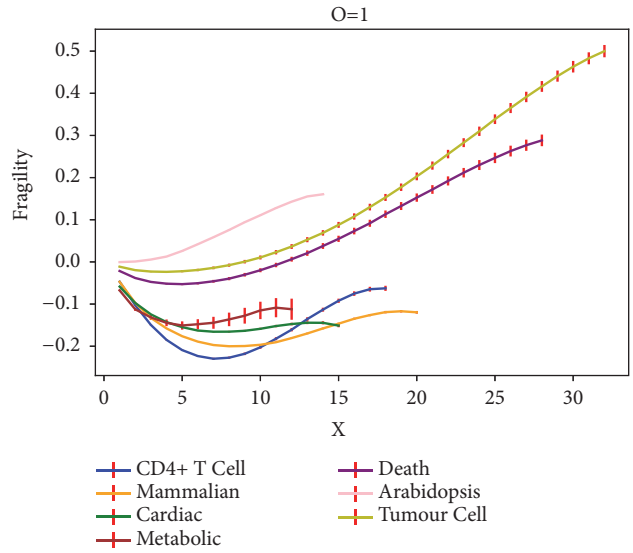


FIGURE 5: ϕ of biological Boolean networks. The error bars represent the standard error of measurements for 1000 different initial states run by 200 steps.

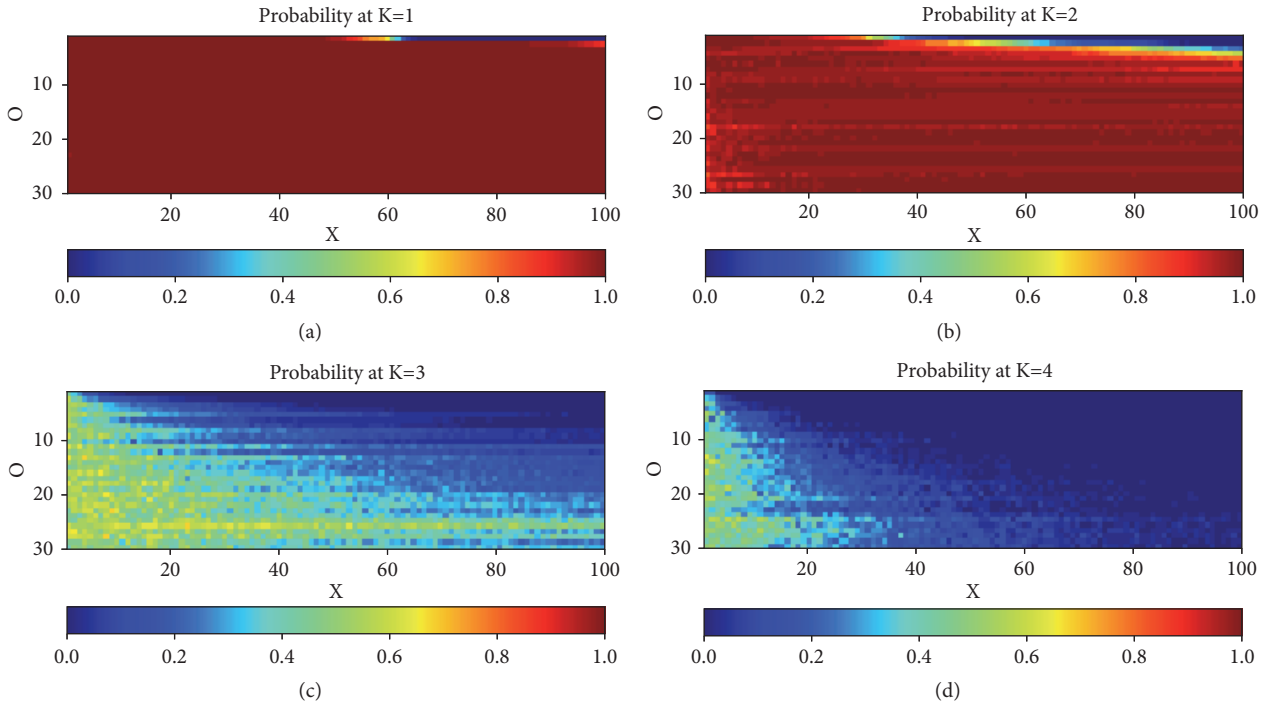


FIGURE 6: Probability of generating antifragile networks depending on X and O for ordered, critical, and chaotic RBNs with $N = 100$, $T = 200$, $p = 0.5$. 50 different networks were used. 10 different initial states were randomly chosen for each network. (a) $K = 1$. (b) $K = 2$. (c) $K = 3$. (d) $K = 4$.

This is consistent with the dynamics of critical RBNs, where criticality is found theoretically at $K = 2$ (when $N \longrightarrow \infty$) and for finite systems at $2 < K < 3$ due to a finite-size effect [25].

However, the result is changed by adding perturbations. In Figure 4(b), the ordered RBNs had the biggest complexity excluding the early range of X, which means that the ordered RBNs show the optimal balance between regularity and change in the presence of noise. This illustrates that systems can exhibit different properties in accordance with the presence of external stressors. Such phenomenon was recently observed in a neural network as well [45], where neural systems showed different dynamical behaviors depending on the presence/absence of external inputs.

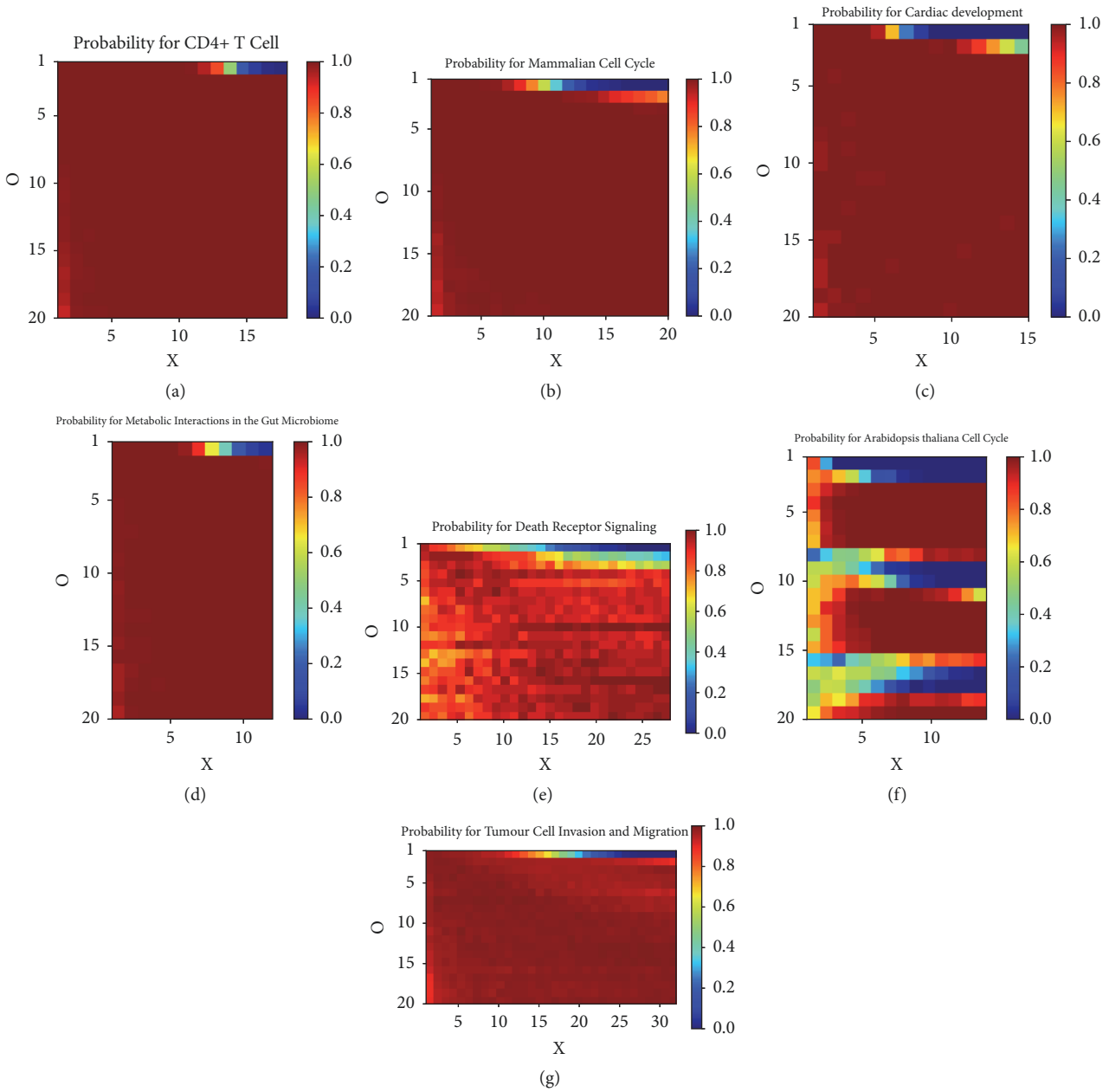


FIGURE 7: Probability of generating antifragile networks depending on X and O for different biological Boolean networks with $T = 200$. 5000 different initial states were used for each network. (a) *CD4+ T cell differentiation and plasticity*. (b) *Mammalian cell-cycle*. (c) *Cardiac development*. (d) *Metabolic interactions in the gut microbiome*. (e) *Death receptor signaling*. (f) *A. thaliana cell-cycle*. (g) *Tumor cell invasion and migration*.

4.2. Antifragility in Biological BNs. Boolean networks have been extensively used as models of genetic or cellular regulation in the fields of computational and systems biology [36–42], because they can capture interesting features of biological systems despite their simplicity. Using seven biological Boolean network models, we measured the values of $\oint$ of biological systems.

We first consider a volatile environment where perturbations are added every time step ($O = 1$). Figure 5 shows that for this high level of noise, the network of *A. thaliana cell-cycle* is fragile, the networks of *death receptor signaling* and *tumor cell invasion and migration* are robust in a certain range of X and fragile in the rest of the range, and the networks of *CD4+ T cell differentiation and plasticity*, *mammalian cell-cycle*, *cardiac development*, and *metabolic interactions in the gut microbiome* are antifragile against perturbations. When comparing with Figure 3(a), we found that antifragility of the biological networks except for *A. thaliana cell-cycle* is similar to that of ordered or critical RBNs.

To obtain more generalized dynamics, we investigated the probability of generating antifragile networks in a diverse range of X and O. Figure 6 is a heat map showing the

probability for RBNs. As shown in the figure, the ordered and critical RBNs can produce antifragile networks. However, if too large perturbations are added in a volatile environment (i.e., $O = 1$), both of them do not exhibit antifragile dynamics. In the case of the chaotic RBNs, they cannot produce antifragile networks in any range of X and O.

Figure 7 is a heat map for the seven BNs. They all show antifragile dynamics like the ordered or critical RBNs. Among the heat maps, the most interesting networks are *A. thaliana cell-cycle* and *CD4+ T cell differentiation and plasticity*. We found that *A. thaliana cell-cycle* repeatedly produces antifragile networks at regular intervals depending on the values of O. Based on many studies demonstrating that living organisms are ordered or critical [46–49], we can infer that *A. thaliana* might have been evolved in environments where particular dimensions of perturbations are added more frequently than other biological systems. We also found that *CD4+ T cell differentiation and plasticity* are the most antifragile of the ones studied, probably because it has the most variable environment. It indicates that our antifragility measure successfully captures the property of the immune system mentioned as a representative example of antifragile systems.

5. Conclusions

In this study, we proposed a new measure of (anti)fragility and applied it to RBNs. Considering an environment given to a system as a noise source, we observed how system properties can be varied depending on the degree of perturbations. We found that ordered and critical RBNs show antifragile dynamics, and especially ordered RBNs are most antifragile against the perturbations. Also, biological systems show antifragile dynamics.

In addition to the findings, we gained a meaningful insight to environments as external stressors. The high complexity with an optimal balance between regularity and change was acquired when moderate perturbations were added very frequently. It means that "optimality" depends on the precise variability of the environment. How can systems be antifragile or robust for varying levels of noise? Which mechanisms can be used to adjust the internal variability depending on the external variability? These questions demand further studies, but possible answers are already being explored based on the results presented here.

Based on the findings and insight, by adjusting the size and frequency of perturbations, we can control system properties from fragile through robust to antifragile dynamics. It may help to understand dynamical behaviors of biological systems depending on environmental conditions and develop new treatment strategies for various diseases including cancer or AIDS, e.g., how can we decrease the antifragility of cancer cells or pathogens? This should reduce their adaptability and potentially improve treatments.

Data Availability

Our simulator and data are available at https://github.com/Okarim1/RBN.git.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Authors' Contributions

Omar K. Pineda and Hyobin Kim contributed equally to this work.

Acknowledgments

We are grateful to Dario Alatorre, Ewan Colman, Luis Ángel Escobar, José Luis Mateos, Dante Pérez, and Fernanda Sánchez-Puig for useful comments and discussions. I would like to acknowledge Dr. Gershenson for his guidance during the MSc course of Adaptive Computation. The final work of the course served as a foundation for this paper [Omar K. Pineda]. This research was partially supported by CONACYT and DGAPA, UNAM.

References

[1] N. Taleb, *Antifragile : Things That Gain from Disorder*, Random House, New York, NY, USA, 2012.

[2] J. Derbyshire and G. Wright, "Preparing for the future: Development of an 'antifragile' methodology that complements scenario planning by omitting causation," *Technological Forecasting & Social Change*, vol. 82, no. 1, pp. 215–225, 2014.

[3] T. Aven, "The concept of antifragility and its implications for the practice of risk analysis," *Risk Analysis*, vol. 35, no. 3, pp. 476–483, 2015.

[4] A. Naji, M. Ghodrat, H. Komaie-Moghaddam, and R. Podgornik, "Asymmetric Coulomb fluids at randomly charged dielectric interfaces: Anti-fragility, overcharging and charge inversion," *The Journal of Chemical Physics*, vol. 141, no. 17, article no. 174704, 2014.

[5] M. Grube, L. Muggia, and C. Gostinčar, "Niches and adaptations of polyextremotolerant black fungi," in *Polyextremophiles*, vol. 27, pp. 551–566, Springer, 2013.

[6] A. Danchin, P. M. Binder, and S. Noria, "Antifragility and tinkering in biology (and in business) flexibility provides an efficient epigenetic way to manage risk," *Gene*, vol. 2, no. 4, pp. 998–1016, 2011.

[7] J. S. Levin, S. P. Brodfuehrer, and W. M. Kroshl, "Detecting antifragile decisions and models: Lessons from a conceptual analysis model of service life extension of aging vehicles," in *Proceedings of the 8th Annual IEEE International Systems Conference, SysCon 2014*, pp. 285–292, Canada, April 2014.

[8] R. Isted, "The use of antifragility heuristics in transport planning," in *Proceedings of Australian Institute of Traffic Planning and Management (AITPM) National Conference*, vol. 3, 2014.

[9] K. H. Jones, "Engineering antifragile systems: A change in design philosophy," *Procedia Computer Science*, vol. 32, pp. 870–875, 2014.

[10] M. Lichtman, M. T. Vondal, T. C. Clancy, and J. H. Reed, "Antifragile communications," *IEEE Systems Journal*, vol. 12, no. 1, pp. 659–670, 2018.

[11] E. Verhulst, "Applying systems and safety engineering principles for antifragility," *Procedia Computer Science*, vol. 32, pp. 842–849, 2014.

[12] C. A. Ramirez and M. Itoh, "An initial approach towards the implementation of human error identification services for antifragile systems," in *Proceedings of the SICE Annual Conference (SICE)*, pp. 2031–2036, September 2014.

[13] A. Abid, M. T. Khemakhem, S. Marzouk et al., "Toward antifragile cloud computing infrastructures," *Procedia Computer Science*, vol. 32, pp. 850–855, 2014.

[14] M. Monperrus, "Principles of antifragile software," in *Proceedings of Companion to the first International Conference on the Art, Science and Engineering of Programming, ACM*, vol. 32, pp. 1–4, Brussels, Belgium, April 2017.

[15] L. Guang, E. Nigussie, J. Plosila, and H. Tenhunen, "Positioning antifragility for clouds on public infrastructures," *Procedia Computer Science*, vol. 32, pp. 856–861, 2014.

[16] R. Serra, M. Villani, A. Barbieri, S. A. Kauffman, and A. Colacci, "On the dynamics of random Boolean networks subject to noise: attractors, ergodic sets and cell types," *Journal of Theoretical Biology*, vol. 265, no. 2, pp. 185–193, 2010.

[17] A. L. Bauer, T. L. Jackson, Y. Jiang, and T. Rohlf, "Receptor cross-talk in angiogenesis: mapping environmental cues to cell phenotype using a stochastic, Boolean signaling network model," *Journal of Theoretical Biology*, vol. 264, no. 3, pp. 838–846, 2010.

[18] M. K. Morris, J. Saez-Rodriguez, P. K. Sorger, and D. A. Lauffenburger, "Logic-based models for the analysis of cell signaling networks," *Biochemistry*, vol. 49, no. 15, pp. 3216–3224, 2010.

[19] R. Albert and H. G. Othmer, "The topology of the regulatory interactions predicts the expression pattern of the segment polarity genes in Drosophila melanogaster," *Journal of Theoretical Biology*, vol. 223, no. 1, pp. 1–18, 2003.

[20] F. Li, T. Long, Y. Lu, Q. Ouyang, and C. Tang, "The yeast cell-cycle network is robustly designed," *Proceedings of the National Acadamy of Sciences of the United States of America*, vol. 101, no. 14, pp. 4781–4786, 2004.

[21] M. Paczuski, K. E. Bassler, and Á. Corral, "Self-organized networks of competing boolean agents," *Physical Review Letters*, vol. 84, no. 14, pp. 3185–3188, 2000.

[22] M. E. J. Newman, "Spread of epidemic disease on networks," *Physical Review E: Statistical, Nonlinear, and Soft Matter Physics*, vol. 66, no. 1, Article ID 016128, 11 pages, 2002.

[23] P. Rämö, S. A. Kauffman, J. Kesseli, and O. Yli-Harja, "Measures for information propagation in Boolean networks," *Physica D: Nonlinear Phenomena*, vol. 227, no. 1, pp. 100–104, 2007.

[24] A. Roli, M. Manfroni, C. Pinciroli, and M. Birattari, "On the design of Boolean network robots," in *Proceedings of the European Conference on the Applications of Evolutionary Computation*, pp. 43–52, Springer, 2011.

[25] N. Fernández, C. Maldonado, and C. Gershenson, "Information measures of complexity, emergence, self-organization, homeostasis, and autopoiesis," in *Guided Self-Organization: Inception*, M. Prokopenko, Ed., vol. 9, pp. 19–51, Springer, 2014.

[26] S. A. Kauffman, "Metabolic stability and epigenesis in randomly constructed genetic nets," *Journal of Theoretical Biology*, vol. 22, no. 3, pp. 437–467, 1969.

[27] S. A. Kauffman, *The Origins of Order Self-Organization and Selection in Evolution*, Oxford University Press, 1993.

[28] C. Gershenson, "Introduction to random Boolean networks," in *Proceedings of the Ninth International Conference on the Simulation and Synthesis of Living Systems (ALife IX)*, pp. 160–173, 2004.

[29] C. Gershenson and N. Fernández, "Complexity and information: Measuring emergence, self-organization, and homeostasis at multiple scales," *Complexity*, vol. 18, no. 2, pp. 29–44, 2012.

[30] C. C. Walker and W. R. Ashby, "On temporal characteristics of behavior in certain complex systems," *Kybernetik*, vol. 3, no. 2, pp. 100–108, 1966.

[31] B. Derrida and Y. Pomeau, "Random networks of automata: A simple annealed approximation," *EPL (Europhysics Letters)*, vol. 1, no. 2, pp. 45–49, 1986.

[32] C. Gershenson, "Guiding the self-organization of random Boolean networks," *Theory in Biosciences*, vol. 131, no. 3, pp. 181–191, 2012.

[33] C. G. Langton, "Computation at the edge of chaos: phase transitions and emergent computation," *Physica D: Nonlinear Phenomena*, vol. 42, no. 1–3, pp. 12–37, 1990.

[34] R. López-Ruiz, H. L. Mancini, and X. Calbet, "A statistical measure of complexity," *Physics Letters A*, vol. 209, no. 5-6, pp. 321–326, 1995.

[35] C. Gershenson, "The sigma profile: A formal tool to study organization and its evolution at multiple scales," *Complexity*, vol. 16, no. 5, pp. 37–44, 2011.

[36] M. E. Martinez-Sanchez, L. Mendoza, C. Villarreal, and E. R. Alvarez-Buylla, "A Minimal regulatory network of extrinsic and intrinsic factors recovers observed patterns of CD4+ T cell differentiation and plasticity," *PLoS Computational Biology*, vol. 11, no. 6, article no. 1004324, 2015.

[37] Ö. Sahin, H. Fröhlich, C. Löbke et al., "Modeling ERBB receptor-regulated G1/S transition to find novel targets for de novo trastuzumab resistance," *BMC Systems Biology*, vol. 3, no. 1, 2009.

[38] F. Herrmann, A. Groß, D. Zhou, H. A. Kestler, and M. Kühl, "A Boolean model of the cardiac gene regulatory network determining first and second heart field identity," *PLoS ONE*, vol. 7, no. 10, article no. 46798, 2012.

[39] S. N. Steinway, M. B. Biggs, T. P. Loughran, J. A. Papin, and R. Albert, "Inference of network dynamics and metabolic interactions in the gut microbiome," *PLoS Computational Biology*, vol. 11, no. 6, article no. 1004338, 2015.

[40] L. Calzone, L. Tournier, S. Fourquet et al., "Mathematical modelling of cell-fate decision in response to death receptor engagement," *PLoS Computational Biology*, vol. 6, no. 3, article no. 1000702, 2010.

[41] E. Ortiz-Gutiérrez, K. García-Cruz, E. Azpeitia et al., "A dynamic gene regulatory network model that recovers the cyclic behavior of arabidopsis thaliana cell cycle," *PLoS Computational Biology*, vol. 11, no. 9, article no. e1004486, 2015.

[42] D. P. Cohen, L. Martignetti, S. Robine et al., "Mathematical modelling of molecular pathways enabling tumour cell invasion and migration," *PLoS Computational Biology*, vol. 11, no. 11, article no. e1004571, 2015.

[43] C. Gershenson and D. Helbing, "When slower is faster," *Complexity*, vol. 21, no. 2, pp. 9–15, 2015.

[44] S. A. Kauffman, *Investigations*, Oxford University Press, 2000.

[45] J. Schuecker, S. Goedeke, and M. Helias, "Optimal sequence memory in driven random networks," *Physical Review X*, vol. 8, no. 4, Article ID 041029, 2018.

[46] I. Shmulevich, S. A. Kauffman, and M. Aldana, "Eukaryotic cells are dynamically ordered or critical but not chaotic," *Proceedings of the National Acadamy of Sciences of the United States of America*, vol. 102, no. 38, pp. 13439–13444, 2005.

[47] E. Balleza, E. R. Alvarez-Buylla, A. Chaos, S. Kauffman, I. Shmulevich, and M. Aldana, "Critical dynamics in genetic regulatory networks: Examples from four kingdoms," *PLoS ONE*, vol. 3, no. 6, article no. 2456, 2008.

[48] M. Villani, L. La Rocca, S. A. Kauffman, and R. Serra, "Dynamical criticality in gene regulatory networks," *Complexity*, vol. 2018, Article ID 5980636, 14 pages, 2018.

[49] B. C. Daniels, H. Kim, D. Moore et al., "Criticality distinguishes the ensemble of biological regulatory networks," *Physical Review Letters*, vol. 121, no. 13, article no. 138102, 2018.

Hindawi
Complexity
Volume 2019, Article ID 8740279, 10 pages
https://doi.org/10.1155/2019/8740279

Research Article

A Boolean Network Approach to Estrogen Transcriptional Regulation

Guillermo de Anda-Jáuregui,[1] Jesús Espinal-Enríquez,[1,2] Santiago Sandoval-Motta,[1,2,3] and Enrique Hernández-Lemus[1,2]

[1]*Computational Genomics Division, National Institute of Genomic Medicine, 14610 México City, Mexico*
[2]*Centro de Ciencias de la Complejidad, Universidad Nacional Autónoma de México, 04510 México City, Mexico*
[3]*Research Chairs Program, National Council on Science and Technology (Conacyt), 03940 México City, Mexico*

Correspondence should be addressed to Enrique Hernández-Lemus; ehernandez@inmegen.gob.mx

Received 1 March 2019; Accepted 6 May 2019; Published 22 May 2019

Academic Editor: Yongtang Shi

Gene expression governs important biological processes such as the cell's growth cycle and its response to environmental signals. Alterations of this complex network of transcriptional interactions often lead to unstable expression states and disease. Estrogen is a sex hormone known for its roles in cell proliferation. Its expression has been involved in several physiological functions such as regulating the menstrual and reproduction cycles in women. Altered expression states where estrogen levels are atypically high have been associated with an increased incidence of breast, ovarian, and cervix cancer. To better understand the implications of deregulation of the estrogen and estrogen receptor regulatory networks, in this work we generated a dynamical model of gene regulation of the estrogen receptor transcription network based on known regulatory interactions. By using an adaptation to classical Boolean Networks dynamics we identified proliferative and antiproliferative gene expression states of the network and also to identify key players that promote these altered states when perturbed. We also modeled how pairwise gene alterations may contribute to shifts between these two proliferative states and found that the coordinated subexpression of E2F1 and SMAD4 is the most important combination in terms of promoting proliferative states in the network.

1. Introduction

1.1. Estrogen Regulation. Estrogen is the primary human female sex hormone, responsible for the development of the female reproductive system and the emergence of secondary sexual characteristics. Estrogens are primarily synthesized in the ovaries but can also be produced in the adrenals and testes. From here, these molecules are distributed to target organs and tissues [1, 2].

To regulate biological processes, estrogen binds to estrogen receptor (ER) proteins. Estrogen may readily diffuse through the cell membrane to the cytosol and bind to ERs. ERs belong to either a nuclear or a membrane class. The nuclear class is comprised of two ERs: alpha and beta. ER-alpha is encoded by the ESR1 gene, located in Chromosome 6(6q25.1-q25.2), meanwhile ER-beta is encoded by the ESR2 gene located in Chromosome 14 (14q23.2-q23.3) [3].

The expression of estrogen receptors is regulated through complex genetic and epigenetic control mechanisms. The most extensively described regulatory mechanisms involve the activity of transcription factor (TF) proteins [4]. These TFs bind to regulatory regions in the genome, inducing changes (either positive or negative) in the basal transcription rate of the genes, ultimately affecting the concentration of the encoded protein.

Estrogen-bound ERs can induce changes in the biological state of a cell through two different mechanisms: a nongenomic and a genomic mechanism. The nongenomic mechanism involves signal transduction through secondary messengers via the Estrogen Signaling pathway and related crosstalk pathways [5]. The genomic mechanism of ERs involves its activity as a transcription factor.

Estrogen-bound ERs translocate to the nucleus where they bind to Estrogen Response Elements located in the

promoter region of estrogen-regulated genes. This binding can either increase or decrease their respective gene transcription rate [6]. We may consider that these two phenomena are not disconnected. ER-regulated genes may be involved in the regulation of other TFs which in turn may regulate ER expression, forming a complex gene regulatory network.

Both experimental and computational studies have been performed to map this gene regulatory network in different biological contexts. The description of the gene regulatory mechanisms governing the expression of ERs, as well as the genes whose expression is regulated through ERs, is still an open problem; however, current knowledge of this regulatory network is enough for us to implement models of the gene regulatory dynamics involving estrogen and estrogen receptor genes.

1.2. Boolean Models of Gene Regulation. Boolean networks have been increasingly used as models for simulating the dynamics of gene regulatory networks. This description treats genes (nodes in the network) as binary variables so genes can only be in two possible expression states: active or inactive. The dynamical behavior of these networks has been proved to give insights into the dynamics of real cellular systems. Boolean networks have been used to model the response of the immune system in response to a respiratory infection [7]. They have served to identify potential therapeutic targets in the blood cancer T cell large granular lymphocyte [8].

Of particular interest in these models are the attractors of the dynamical system, defined as the set of recurrent states that the networks reach after some time [9]. The dynamics of the network are simulated by determining the state of each element of the network at a posterior time, given the current state of its regulators. The order by which each state of each element is updated can be either synchronous or asynchronous (see Appendix), and thus the system can be either Markovian of purely deterministic.

Since Boolean networks are discrete dynamical systems with finite support (there are exactly 2N possible states on a Boolean network with N nodes), the evolution of the system will produce recurrent states. The trajectories will fall into one of a set of steady states or cycles called attractors. The collection of attractors along with the respective configurations that lead to them is called the attractor landscape.

The determination of the set of attractor states and the convergence dynamics leading to those attractors constitutes the solution to the Boolean network dynamics problem. It is important to note that these attractors are often consistent with what is observed experimentally in the gene expression patterns of specific phenotypes or cell types [9–11].

In this paper we present a Boolean model for the estrogen and estrogen receptor gene regulatory network for which we analyze its dynamics under different configurations of presence / absence of nodes and determine which configurations lead to a proclivity to proliferative and antiproliferative states.

2. Analysis

2.1. Construction of the Estrogen Receptor Regulatory Network. To study the gene regulatory phenomenon surrounding the estrogen receptor using the Boolean formalism, a suitable network is required. The reconstruction of such regulatory network is not trivial and will often represent a fragment of the whole set of gene regulatory interactions that could be involved in real organisms. However, taking into account only the first-order regulators often represents a good first approach model to understand how the network behaves under different circumstances [8, 12].

The problem may be approached using much formalism, including coexpression and information-theoretical methods [13], literature mining strategies [14], and even reconstruction using the Boolean formalism itself [15]. Such approaches may become quite computationally expensive, and the reconstructed network itself must be validated against known bibliographic information. With these considerations in mind, we decided to construct a background network for our Boolean dynamics based on well-documented, curated information of transcriptional regulations available in databases.

We constructed this network of gene regulation centered on estrogen receptors α and β (ESR1 and ESR2). We identify regulatory interactions previously described and validated in the current biomedical literature. For this and in order to increase the reliability of the constructed network, we made use of two complementary databases of regulatory interactions: RegNetwork and IPA,

Our first source of regulatory interactions, RegNetwork [16] is a database of transcriptional regulatory relationships in human. This database comprises curated regulations from various other databases (BioGrid, FANTOM, HPRD, IntAct, JASPAR, TRANSFAC, TRED, among others.) that integrate the current understanding of regulatory interactions derived from experimental and theoretical data, for instance by resorting to gene expression datasets for known gene perturbations, and ChIP-seq assays. The database assigns different confidence levels to the listed interactions based on the information provided by each of its sources. Furthermore, it makes a distinction between predicted and experimentally validated regulatory interactions.

Our second source of regulatory interactions is the Ingenuity Pathway Analysis (IPA) platform (IPA ©, QIAGEN Redwood City). A commercial software package that relies on a proprietary Ingenuity Knowledge Base [17] which contains a causal network derived from experimental observations as well as records from manually curated biomedical literature.

Using these two sources, we constructed our *estrogen receptor regulatory network* following these steps:

(1) Extract first neighbors of ESR1 and ESR2 from RegNetwork; consider only interactions with experimental evidence and high confidence.

(2) Extract interactions between first neighbors of ESR1 and ESR2, considering only interactions with experimental evidence and high confidence. This will be Network 1.

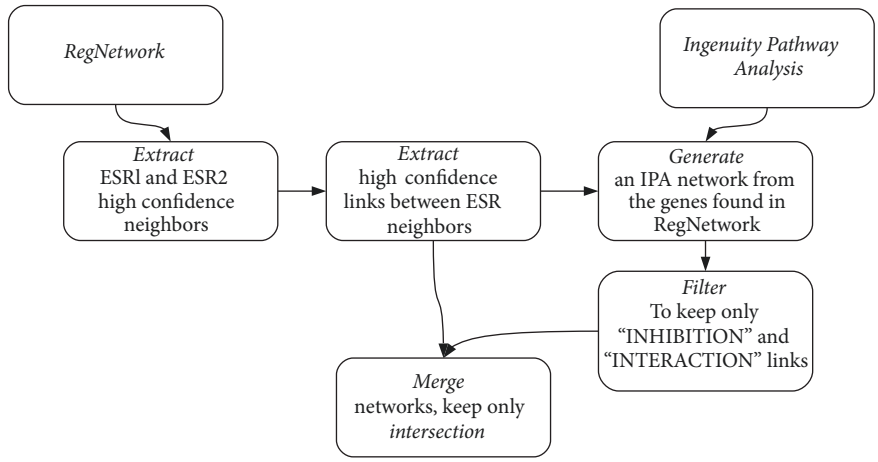


Figure 1: Pipeline followed in this work.

(3) Filter out nodes in Network 1 with $out - degree = 0$ iteratively, until there are no nodes with $out-degree = 0$. This will be Network 2.

(4) Take the nodes in Network 2, and use them as the input of an IPA analysis to generate networks based on interactions described in IPA. We merge all IPA networks and then remove nodes that were not present in Network 2 to generate Network 3.

(5) We filter Network 3 by keeping only those links that are described by IPA as being either "INHIBITION" or "ACTIVATION," and we remove all nodes with $out - degree = 0$ iteratively until there are no nodes with $out - degree = 0$. This generates Network 4.

(6) We generate a network with the intersection of nodes and edges of network 2 and network 4 and iteratively remove nodes with $out - degree = 0$. This generates the final *estrogen receptor regulatory network* to be used in this work (Figure 1).

It is known that the general inference of this kind of networks is a computationally expensive problem (NP-hard). By considering a limited search space as we just sketched, we are able to directly delve into the phenomenological implications of the studied system. This is so since fixing the network in compliance to established experimental facts and known biological information by resorting to causal inference has proven to be an effective way to proceed, in particular in cases like the estrogen pathway, for which a large number of experimental evidence sources are available.

2.2. Boolean Rules and Dynamics. Boolean rules are the set of logical constraints that a node may have in a Boolean network. This rule depends on the state of regulators of the node, and the activatory/inhibitory nature of those regulators. Let us explain one dynamic rule of this signaling pathway. The estrogen receptor 1, ESR1, is a well-known gene that encodes a protein which participates in processes such as DNA binding, activation of transcription, or sexual development. This protein (among other functions) binds to the androgen receptor (AR), which in turn regulates negatively the expression of ESR1. STAT5A is another transcription factor that promotes ESR1 activity. At the same time, BRCA1, a crucial gene in DNA damage response, regulates positively the activity of ESR1.

Table 1: ESR1 truth table.

AR(t)	BRCA1(t)	STAT5A(t)	$ESR1(t+1)$
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

ESR1 has two positive regulators (AR and STAT5A) and one repressor (BRCA1). The dynamical state of ESR1 at time $t + 1$, will depend on the state of AR, BRCA1, and STAT5A at time t. The logical rule of ESR1 may be written as in Table 1

In this case, the single or combined action of BRCA and STAT5A will activate the estrogen receptor ($ESR1 = 1$) only if AR is not present; otherwise, ESR1 will acquire a value of 0. Regarding the combined regulation of ESR1, as in the rest of this dynamical system, the negative regulators exert a stronger influence than the positive ones. This differentiated influence of negative regulators has been applied in other biological Boolean networks before [9–11], since in some cases, it is not possible to know experimentally the effect of combined action of more than one gene/protein over a determined molecule. It has been shown that the differentiated action of negative and positive regulators is often in agreement with the biological system.

2.3. Boolean Model and Attractor Landscape Analysis. Our simplified estrogen receptor regulatory network consists of 14 nodes (Figure 2). In this dynamic model, each node can

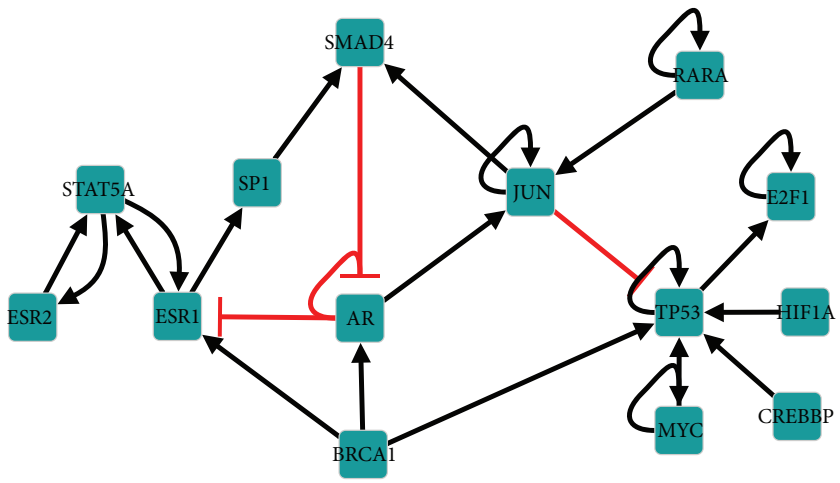


FIGURE 2: The estrogen receptor regulatory network. It was constructed from curated transcriptional interactions found in RegNetwork and IPA. Gene nodes include ESR1 and ESR2, the estrogen receptors. It has 25 interactions representing transcriptional regulation among these genes. Inhibitory regulation is represented in red; activatory regulation is depicted in black.

acquire a set of discrete values that correspond to its possible expression levels. Due to the lack of experimental data on the kinetic constants for each interaction of the network, we constructed a model that focuses on the functional state of expression of each component, rather than on their exact concentrations. These levels of expression are modeled through discrete variables that take a finite number of values. Since all the elements of the network are considered binary, our network has fourteen binary nodes giving a total of $\Omega = 2^{14}$ possible dynamical states for the network.

At every time step of the dynamics, the expression level of all components of the network is updated simultaneously according to

$$\sigma_n(t+1) = F_n\left(\sigma_{1n}(t), \sigma_{2n}(t), \ldots, \sigma_{kn}(t)\right) \quad (1)$$

where $\sigma_n(t)$ represents the state of the n^{th} element of the network at time t. Here $\{\sigma_{1n}, \sigma_{2n}, \ldots, \sigma_{kn}\}$ are the k_n regulators of σ_n and $F_n(\cdot)$ is a discrete function, stated as a logical rule, that explicitly states the corresponding expression level of σ_n given the current expression levels of its regulators. $F_n(\cdot)$ is constructed according to experimental evidence regarding the regulatory interactions (activator or inhibitor) for each node (see methods for references). All the functions $F_n(\cdot)$ for the estrogen network are listed in Supplementary Information 1.

Since there is a finite number of possible dynamical configurations for the entire network (Ω), starting the dynamics from any of these configurations and successively iterating (1) for each node will make the network traverse through a series of states until a periodic pattern of activity is reached. This periodic set of states is known as an attractor.

Several attractors might exist for a given network, and several initial configurations may lead to the same attractor. For a fixed set of logical functions $F_n()$, the particular attractor the network falls into depends entirely on the initial condition the network starts from. Each attractor has a basin of attraction defined by all of the initial conditions that lead to that particular attractor. These attractors can be thought of as stable patterns of activity of real biological systems as has been shown previously [8, 9]. In this case, the attractors reached will represent the proliferative or antiproliferative state in cells under the transcription signals triggered by estrogen.

Despite these parallelisms, a direct comparison of an attractor to real expression levels might not be so straightforward. Cyclic attractors (attractors composed by several states) are very common whereas experimental gene expression is often presented as a single value. Additionally, gene expression measurements are commonly taken from cell populations, which makes the final measurement an average of single measurements taken over a time window. For this reason we have used a modification of the classic Boolean Network approach where we average the gene expression in the whole attractor basin to end up with a single level of expression for each gene. This approach has been used previously to accurately simulate the behavior of small gene regulatory networks [12]

Here the state of each element of the network is represented by its average expression over a time lapse. The length of the window (L) where values will be averaged corresponds to the length of the attractor reached. For instance, let us assume a cyclic attractor consisting in four network states is reached. For simplicity let us assume the states are

State (1) $[1]_1[0]_2[0]_3[0]_4[0]_5[0]_6[0]_7[0]_8[0]_9[0]_{10}[0]_{11}[0]_{12}[0]_{13}[0]_{14}$,

State (2) $[0]_1[0]_2[0]_3[0]_4[0]_5[0]_6[0]_7[0]_8[0]_9[0]_{10}[0]_{11}[0]_{12}[0]_{13}[0]_{14}$,

State (3) $[1]_1[1]_2[0]_3[0]_4[0]_5[0]_6[0]_7[0]_8[0]_9[0]_{10}[0]_{11}[0]_{12}[0]_{13}[0]_{14}$,

State (4) $[0]_1[1]_2[0]_3[0]_4[0]_5[0]_6[0]_7[0]_8[0]_9[0]_{10}[0]_{11}[0]_{12}[0]_{13}[0]_{14}$.

Now, we want the averaged state of $[node]_1$, which is represented by the first digit of the network state

([1], [0], [1], [0]). Then the corresponding expression level for $node_1$ for this attractor will be $(1 + 0 + 1 + 0)/4 = 0.5$ where 4 is the size of the attractor.

Since a network can have more than one attractor, we will end up with an expression level for each of these attractors. In order to account for each of their basins of attraction and end up with a single value for the expression level of a gene, we have incorporated a weighted average using the entire set of attractors (N) for each network. We define the average expression level of σ_n as

$$\sigma_n = \sum_{a=1}^{N} \omega_a \left(\frac{\sum_{\tau}^{L_a} \sigma_n(\tau)}{L_a} \right) \quad (2)$$

where N represents all the different attractors, so the external sum is carried out over all the existing attractors numbered 1 to N. The parameter ω_a is the fraction of initial conditions that lead to attractor "a" over the total possible conditions, that is, the size of the basin of the a_{th} attractor over Ω. The internal sum is carried out over all the $\sigma_n(\tau)$ states of the a_{th} attractor, of size L_a. This means that the final level of expression of the gene σ_n will be the sum of the averaged expression levels of σ_n in each attractor, with each attractor average weighted by its corresponding basin of attraction.

This modification, apart from allowing an easier comparison between the model and experimental data, resembles the way in which experimental data is gathered for gene expression, where traditionally measurements of the level of expression represent the population average, as cells in the population may be at different stages of a stable pattern of gene expression.

It is important to note that mutations in our model (e.g., a deletion/malfunction of a gene) are represented by keeping the value of the deleted node equal to zero throughout all the dynamics. In the case of gene overexpression, we kept the value of the overexpressed gene equal to its maximum state over the whole simulation.

2.4. Perturbation Analysis. To simulate altered physiological states, we explored the dynamics of the network if the state of genes is perturbed.

(i) We simulated the *overexpression* of a gene by setting it to an *ON* state at the beginning of the simulation and keeping it that way throughout the simulation regardless of the state of its regulators.

(ii) We simulated the *knockout* of a gene by setting it to an *OFF* state at the beginning of the simulation and keeping it that way throughout the simulation regardless of the state of its regulators.

The simulation of the overexpression of a gene may represent either an increase in the activity or concentration of said gene in a phenotype and may also be used as a model of the activity of an agonist drug. Similarly, the simulation of the knockout of a gene may also be used as a model of the activity of an antagonist drug.

We analyzed the full set of single *overexpression* and *knockout* perturbations for all genes in the network. We also analyzed the full set of 2-hit perturbations for all genes in the network, including all *overexpression/overexpression*, *overexpression/knockout*, and *knockout/knockout* pairs. This has been conducted successfully in other biological systems by members of our group, such as the calcium-dependent signaling pathway of the spermatozoa of the sea urchin *S. purpuratus*, in its searching for the egg [10].

2.5. Proliferation Index. We used the Boolean dynamics to quantify biological features in a particular phenotype. In this work, we focused on identifying whether a given phenotype may tend to be proliferative or antiproliferative. To do this, we considered whether the expression of each gene may be involved in processes that are proliferative or antiproliferative, beyond their role as regulators in the network.

We constructed a *Proliferation Index* (PI), in which we consider, for a given phenotype, the average state of each gene throughout the attractor landscapes associated to the phenotype.

$$PI = \frac{\sum \langle P \rangle - \sum \langle AP \rangle}{\left\langle \sum \langle P \rangle + \sum \langle AP \rangle \right\rangle} \quad (3)$$

Where P are proliferative genes and AP are antiproliferative genes and $\langle \cdot \rangle$ are the appropriate ensemble averages. In order to assess whether a gene was considered *proliferative* or *anti-proliferative*, we performed a systematic analysis of the literature using a combination of Pubmed https://www.ncbi.nlm.nih.gov/pubmed/, the Gene database [18], and the Genetics Home Reference https://ghr.nlm.nih.gov/. In Supplementary Information 2 we provide the bibliographic evidence used to asign a *proliferative* or *anti-proliferative* value to each gene in the network. Our index should not be confused with the proliferative index (or growth fraction), which is used in the clinical setting.

It is worth to mention that the Proliferation Index (PI) defined here is the result of averaging the state value of nodes during the attractor period. The PI is a measure that integrates the attractor landscape in terms of the proliferative/antiproliferative phenotype.

3. Results

3.1. The Estrogen Transcriptional Network. By following our construction methodology, we are able to recover an *estrogen receptor regulatory network* composed of 14 nodes and 25 directed and signed interactions. Four of these interactions are inhibitory, while the rest correspond to activation. A visualization of this network may be found in Figure 2.

The network dynamics of this network is depicted in Figure 3, where each dynamical state is represented as a colored point, and the transition between two consecutive states is represented as a straight line. As previously mentioned, attractors of the network dynamics may be punctual or cyclic. In the figure we observe both cases.

3.2. Effects of Perturbations on Proliferation Based on Network Dynamics. Through the use of well-curated biological knowledge along with Boolean network dynamics, we

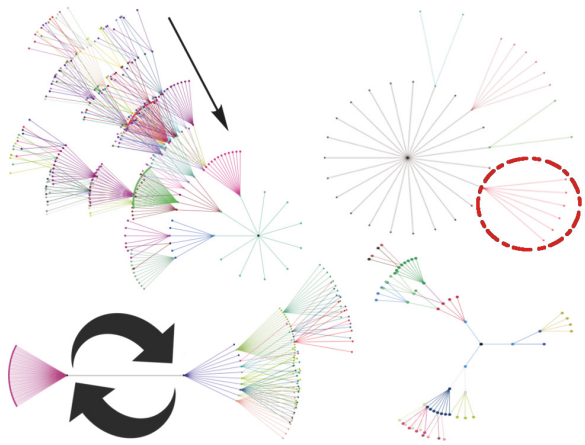


Figure 3: Excerpt of the attractor landscape. Fan-like representation of a set of 4 attractors of the attractor landscape of the estrogen/estrogen receptor network. Each point represents a state of the network. Connections represent temporal succession between states, with the outward points representing preceding states (see black arrow). Same colored fans represent a collection of states that lead to the same future state of the network (e.g., red dashed circle). A cyclic attractor of size = 2 is also represented (see blue and purple circular arrows). The length of the connections is inconsequential as all time steps between states are fixed.

developed a model that may elucidate the contribution of gene perturbations to an observable phenotypic trait. We focused on the proliferative state that is achieved through gene perturbation. This could reflect the changes in cell growth observed in diseases such as cancer, but it also can be used to model the effects that an external perturbation (such as a pharmacological intervention) may have in the phenotype.

3.2.1. The Effects of Single Perturbations on the Proliferative Phenotype. In Figure 4 we present the result of the perturbation of single genes in terms of the *Proliferation Index* (*PI*), compared to the PI value for the wild-type (*WT*) phenotype.

A total of 28 perturbations were performed, which may be seen in Figure 4(a). Overall, 15 of these perturbations induce PI value higher than the one for the WT phenotype ($PI = -0.0761$), whereas 13 lead to a reduction of the PI value with respect to the WT. The maximum PI value is achieved through the knockout of E2F1 ($PI = 0.3029$), while the minimum PI value is achieved through the knockout of STAT5A ($PI = -0.5600$).

We may observe that the effects of gene overexpression and knockout are different in terms of the PI. In Figures 4(b) and 4(c) we may observe the PI for overexpressions and knockouts separately. In the case of overexpressions (Figure 4(b)), the PI values are less spread, ranging from −0.3670 to 0.0335 with 9 perturbations increasing the PI with respect to the WT and 5 decreasing it. In the case of knockouts (Figure 4(c)), these cover a broader range including the aforementioned overall maximum (E2F1 knockout) and minimum (STAT5A knockout); 4 perturbations increase the PI with respect to WT, and 10 decrease it.

In Figure 4(d) we present the genes in the network in a scatterplot, where the x-axis represents PI when the gene is knocked-out, and the y-axis represents PI when the gene is overexpressed. We trace four quadrants with respect to PI for the WT phenotype. We may observe that all four antiproliferative genes are placed in the lower, right quadrant, indicating that their knockouts lead to more proliferation, while their overexpression leads to less proliferation.

3.2.2. Effects of Two-Hit Perturbations on the Proliferative Phenotype. In Figure 5 we present the results of the simultaneous perturbation of two genes of the Estrogen Receptor Regulatory Network in terms of the Proliferative Index as heatmaps. In Figure 5(a), we present the result of the simultaneous overexpression of two genes. In Figure 5(b), we show the effect of the combined overexpression of a gene (shown in the rows of the heatmap) and the knockdown of another gene (shown in the columns of the heatmap). Finally, in Figure 5(c), we show the effect of double gene knockouts.

For each type of two-hit perturbation, we may find a maximum and minimum PI value. In the case of the double overexpression, the minimum PI value is achieved with the overexpression of TP53 and AR ($PI = -0.5313$), while the maximum PI value is achieved with the overexpression of ESR1 and AR ($PI = 0.1876$). For overexpression/knockout combinations, the minimum PI value results from the overexpression of TP53 and the knockout of STAT5A($PI = -0.8871$), while the maximum PI value comes from overexpressing ESR1 and knocking out SMAD4 ($PI = 0.4719$). In the case of double knockouts, knocking out both STAT5A and JUN leade to the minimum PI value ($PI = -1.0196$) while the double knockout of SMAD4 and E2F1 generates the maximum PI value ($PI = 0.8033$)

Through the double perturbation of genes, it is possible to reach more extreme changes in PI than by targeting a single gene alone. For instance, the lowest PI value obtained ($PI = -1.0196$, from the double knockout of STAT5A and JUN) is much lower than the lowest PI obtained from a single gene perturbation ($PI = -0.5600$ from the single knockout of STAT5A). Similarly, the highest PI value obtained ($PI = 0.8033$ from the double knockout of SMAD4 and E2F1) is higher than the highest PI value obtained from single perturbations ($PI = 0.3029$ from the knockout of E2F1). Importantly, and similar to what was observed in single perturbations, the most extreme changes in PI come from knockout perturbations.

4. Discussion

We have shown that with the Boolean approach it is possible to perturb the dynamical state of the estrogen transcriptional network and observe single or multitarget perturbations. As it is expected, the effect of altering one or more elements in the network dynamics will be different in terms of the Proliferation Index.

In the upper, left quadrant of the scatterplot in Figure 4(d), representing single gene perturbations, we

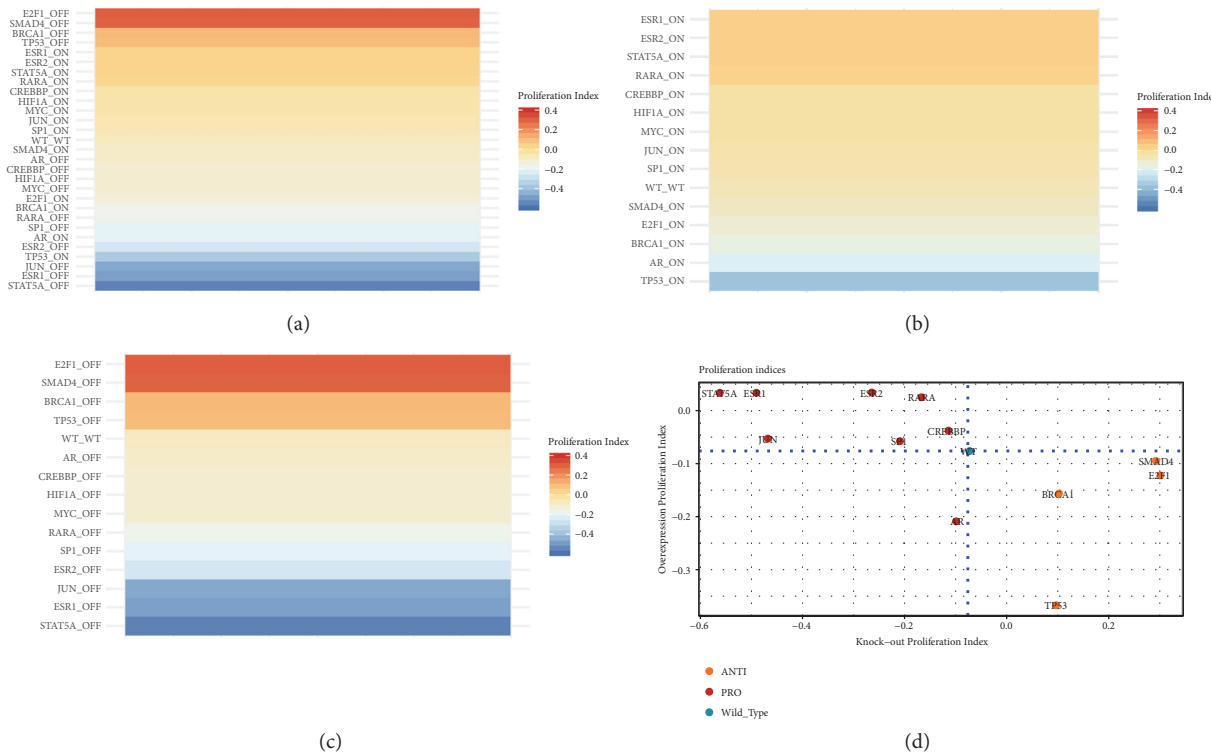


Figure 4: Proliferation indexes for single gene perturbations. Each column shows the PI values after a perturbation. (a) shows the set of overexpression and knockouts, (b,c) represent overexpression and knockout separately. (d) is a scatterplot showing the PI values of all genes after overexpressing and knocking out the genes.

may find proliferative genes (STAT5A, ESR2, MYC, JUN, etc.), meaning the overexpression of these genes lead to more proliferation, while their knockout leads to less proliferation. An interesting finding is the curious case of AR. This is the only gene in the network that is located in the lower left quadrant, indicating that both its overexpression and knockout lead to a decrease of the proliferative index with respect to the wild type.

The E2F1 gene is a well known tumor suppressor gene. It participates in both control of cell cycle and cell death processes. It has been observed experimentally that lower expression values of E2F1 gene are frequent in malignant tumors in breast cancer [19]. As in our network dynamics, the highest *PI* value was obtained by knocking out E2F1 gene, which is in agreement with the experimental results. Analogously, as STAT5A, being one of the main activators of ESR1 and ESR2, its inhibition decreases substantially *PI*.

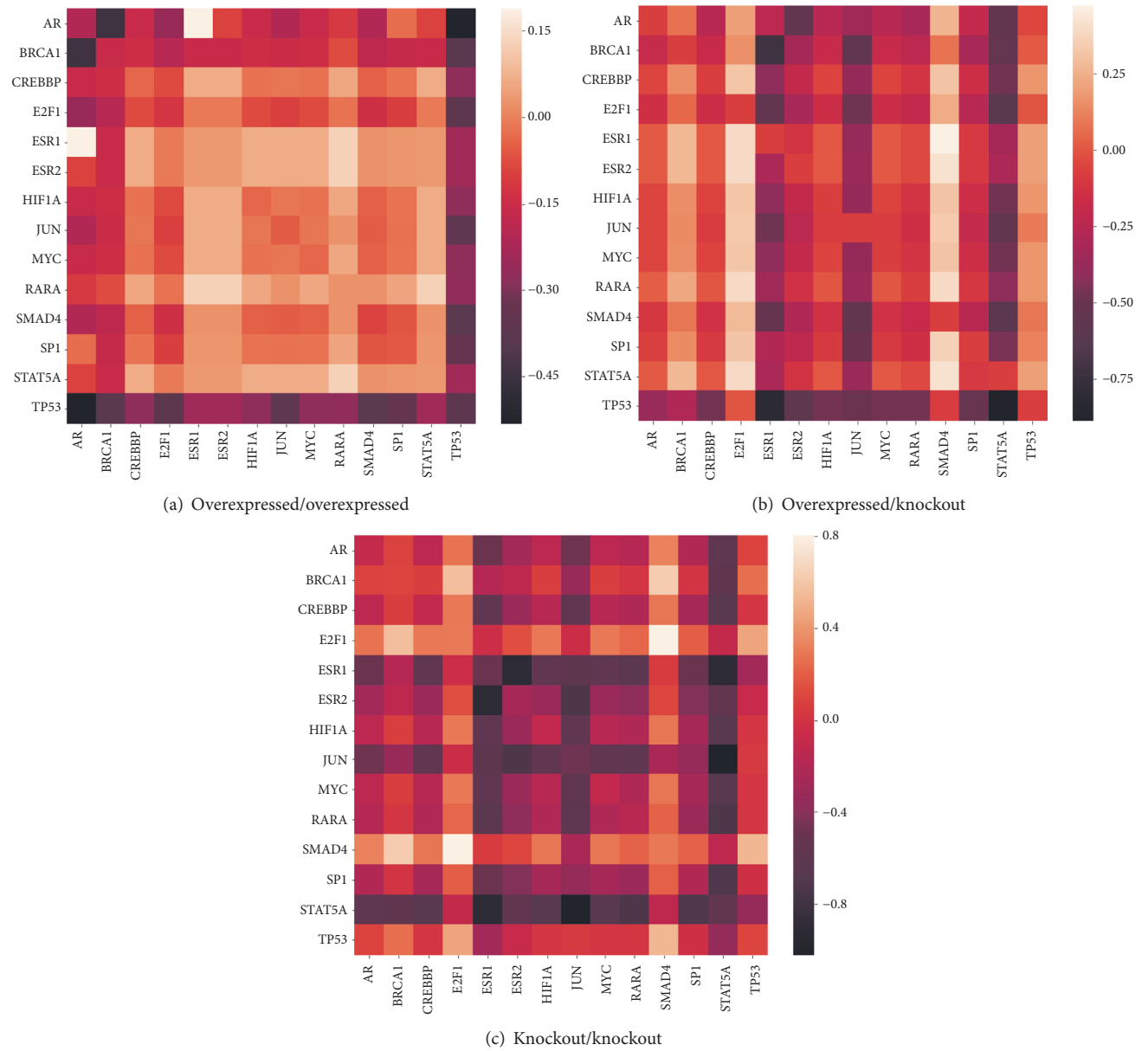
By observing Figure 5, representing two-hit perturbations, it is evident that each type of perturbation generates different clustering patterns. It may be seen that in the case of double overexpressions, we observe a more homogeneous distribution of the PI values. In the case of the overexpression/knockout combinations, the PI patterns tend to be more dominated by the knockout genes (as the pattern observed is of vertical stripes). Finally, in the case of double knockouts, we may find well defined clusters that are related to the double knockout of antiproliferative genes, proliferative genes, or the combination of a proliferative and antiproliferative gene.

It is worth noting that the lowest PI value results from the concerted action of the overexpression of a keystone tumor suppressor (TP53) and the concomitant knockout of a proproliferative gene (STAT5A), whose single knockout leads to the lowest individual PI value. The aforementioned results may have important implications in different scenarios, such as cancer, where drug combinations may have deep impact in clinical outcomes.

Interestingly, the resulting network, after causal inference, contains Master Regulators such as P53, E2F, SMAD4, STAT5A, AR, ESR1, MYC, FOS, or JUN. It is well known that these Master Regulators determine the cell phenotype in health and disease and its deregulation may have profound implications, in cases such as cancer [20].

PI was constructed acknowledging the pro- and antiproliferative activity of said regulators. The relevance of having Master Regulators in our network is that the "fine tuning" of them would imply the switch to a proliferative state or a cell cycle arrest.

The Boolean approach used here has several advantages, such as the fast and direct set of results that are obtained by a relatively simple model. It is not necessary to know the reaction rates or other biological parameters that are often difficult to obtain experimentally. Perturbation analysis is also easy to obtain and interpret. Another advantage is the possibility to perturb more than one molecule *in silico* and analyze results in terms of transient times, attractor landscapes, or basins of attraction.



(a) Overexpressed/overexpressed

(b) Overexpressed/knockout

(c) Knockout/knockout

FIGURE 5: Proliferation indexes for two-hit perturbations. The three heatmaps show the PI value for perturbation of a couple of genes in the network.

However, these kinds of models also present some issues that must be taken into account to have a better interpretation: the model only uses two discrete states, loosing the fine-tuning of studying the system as a continuous model. Time evolution is also discrete, but it is widely known that biological molecules have a particular time for reaction. Despite the fact that the Boolean model uses a discrete time evolution, this does not significantly differ from an attractor landscape obtained by a nonsynchronous update evolution dynamics.

All of these caveats obviously have influence on the interpretation, but after a careful construction of the dynamical rules of the network, the results of the Boolean dynamics are a good generator of hypotheses and may be used as a first step in the searching for experimental corroborations.

5. Conclusions

In this work we have demonstrated *in silico* that altering the dynamical state of key biomolecules of the proliferative estrogen regulatory networks is possible to shift the dynamical state from a proproliferative towards an antiproliferative one and vice versa. The Proliferation Index presented here, despite being similar to other indexes used in cancer-related Boolean networks [21], provides elements of analysis and suggests possible experimental approaches in terms of altering the estrogen-dependent cell proliferation.

These kinds of approaches may be useful to test the usage of different drugs with a known or unknown effects and evaluate the final outcome searching for a more personalized medicine.

Appendix

Boolean Networks as Dynamical Systems

This brief appendix provides some definitions of dynamical systems and Boolean networks included for the sake of completeness. The study of the dynamical evolution of networked systems has been gaining importance and recognition in the physics/applied mathematics/complex systems/computer sciences literature. This is so since a wide variety of nontrivial phenomena has been characterized as arising of the dynamic evolution of interdependent agents. Features like cooperation, spreading, and synchronization dynamics on networks have been characterized. For instance, the work of Wang and coworkers [22] presents an application of novel centrality measures to account for modified diffusion (spreading) on complex networks while information sharing and cooperation have been characterized in the works of the Chengyi group [23, 24].

Boolean networks, in particular, are a class of (deterministic or stochastic) sequential dynamical systems. Boolean networks usually consist on a (finite) set of Boolean logic variables governed by a set of finitary functions of the form $\mathscr{F} : \mathrm{B}^k \longrightarrow \mathrm{B}$, where $\mathrm{B} = \{0, 1\}$ is a binary logic or *Boolean domain* (e.g., an algebra of logical truth values) (it is possible to build Boolean domains with more than two logical states. The formalism extension to these cases is straightforward.) and k is the *arity* (number of arguments or Cartesian product dimension), a nonnegative integer. A Boolean function is thus a propositional formula in k variables which takes a series of inputs from a subset of the Boolean variables and as an output produces the state of the corresponding variable. The set of Boolean functions determines the connectivity of the set of variables that become the nodes of a network, whose topology is given by the combination of Boolean functions for all the variables [25, 26].

For the Boolean network dynamics, the state of the network at a given time $t + 1$ is determined via the evaluation of each of the variables' function on the state of the network at a previous time t. This may be done on a *synchronous* (all nodes' states updated at once) or *asynchronous* (hierarchical updating given the position of a given node in the network) way. Depending upon updating procedures, the system's dynamics may be Markovian or non-Markovian (often finite-Markovian) [26, 27].

Given the fact that Boolean networks are discrete dynamical systems with finite support (there are exactly 2^N possible states on a classical—i.e., 2-state—Boolean network with N nodes), the evolution of the system will produce recurrent states. The trajectories will fall into one of a set of steady states or cycles called *attractors*. The set of attractors of a dynamical system is called the *attractor landscape*. The determination of the set of attractor states and the convergence dynamics leading to those attractors constitutes the solution to the Boolean network dynamics problem [27].

The Boolean networks studied here belong to a class of deterministic dynamical systems. Such systems may be represented by a set of differential equations describing the dynamical evolution in phase space. Deterministic Boolean networks may also be represented as a discrete dynamical system (a map) that when iterated reproduces the full dynamics of the network including the set of attractors. This was the way we proceeded here. Since iterated maps and differential equations are two equivalent representations of the evolution of a dynamical system [28], our approach does not loose any generality.

Data Availability

All relevant data has been included in the supplementary materials.

Conflicts of Interest

The authors have no conflict of interest to declare.

Authors' Contributions

Guillermo de Anda-Jáuregui, Jesús Espinal-Enríquez, and Santiago Sandoval-Motta contributed equally to this work.

Acknowledgments

The research leading to these results has received funding from Consejo Nacional de Ciencia y Tecnología (grant number 285544/2016, Ciencia Básica, and 2115/2017, Fronteras de la Ciencia (Jesús Espinal-Enríquez)), as well as federal funding from the National Institute of Genomic Medicine (Enrique Hernández-Lemus). Enrique Hernández-Lemus also acknowledges support from the 2016 Marcos Moshinsky Research Chair in the Physical Sciences. Jesús Espinal-Enríquez acknowledges support from Fundación Miguel Alemán in Health Research. Santiago Sandoval-Motta acknowledges support from the program Cátedras CONACYT. The funders had no role in the design of this research.

Supplementary Materials

Supplementary 1. Supplementary Material 1. Regulatory functions of the estrogen transcriptional networks. Each file contains the regulatory function for all those genes in the network, including the regulatory genes, as well as the discrete value of the target gene after taking into account the value of its regulators.

Supplementary 2. Supplementary Material 2. Bibliographic evidence associated with the proliferative and antiproliferative nature of the genes in the network.

References

[1] W. E. Stumpf, "Nuclear concentration of 3H-estradiol in target tissues. Dry-mount autoradiography of vagina, oviduct, ovary, testis, mammary tumor, liver and adrenal.," *Endocrinology*, vol. 85, no. 1, pp. 31–37, 1969.

[2] J. Cui, Y. Shen, and R. Li, "Estrogen synthesis and signaling pathways during aging: from periphery to brain," *Trends in Molecular Medicine*, vol. 19, no. 3, pp. 197–209, 2013.

[3] F. Pedeutour, B. J. Quade, S. Weremowicz, P. Dal Cin, S. Ali, and C. C. Morton, "Localization and expression of the human estrogen receptor beta gene in uterine leiomyomata," *Genes, Chromosomes and Cancer*, vol. 23, no. 4, pp. 361–366, 1998.

[4] L. Giacinti, P. P. Claudio, M. Lopez, and A. Giordano, "Epigenetic information and estrogen receptor alpha expression in breast cancer," *The Oncologist*, vol. 11, no. 1, pp. 1–8, 2006.

[5] G. De Anda-Jáuregui, R. A. Mejía-Pedroza, J. Espinal-Enríquez, and E. Hernández-Lemus, "Crosstalk events in the estrogen signaling pathway may affect tamoxifen efficacy in breast cancer molecular subtypes," *Computational Biology and Chemistry*, vol. 59, pp. 42–54, 2015.

[6] P. Ascenzi, A. Bocedi, and M. Marino, "Structure-function relationship of estrogen receptor α and β: Impact on human health," *Molecular Aspects of Medicine*, vol. 27, no. 4, pp. 299–402, 2006.

[7] J. Thakar, M. Pilione, G. Kirimanjeswara, E. T. Harvill, and R. Albert, "Modeling systems-level regulation of host immune responses," *PLoS Computational Biology*, vol. 3, no. 6, Article ID e109, 2007.

[8] A. Saadatpour, R.-S. Wang, A. Liao et al., "Dynamical and structural analysis of a t cell survival network identifies novel candidate therapeutic targets for large granular lymphocyte leukemia," *PLoS Computational Biology*, vol. 7, no. 11, Article ID e1002267, 2011.

[9] J. Espinal, M. Aldana, A. Guerrero, C. Wood, A. Darszon, and G. Martínez-Mekler, "Discrete dynamics model for the speract-activated Ca 2+ signaling network relevant to sperm motility," *PLoS ONE*, vol. 6, no. 8, Article ID e22619, 2011.

[10] J. Espinal-Enríquez, A. Darszon, A. Guerrero, and G. Martínez-Mekler, "In Silico determination of the effect of multi-target drugs on calcium dynamics signaling network underlying sea urchin spermatozoa motility," *PLoS ONE*, vol. 9, no. 8, Article ID e104451, 2014.

[11] J. Espinal-Enríquez, D. A. Priego-Espinosa, A. Darszon, C. Beltrán, and G. Martínez-Mekler, "Network model predicts that CatSper is the main Ca2+ channel in the regulation of sea urchin sperm motility," *Scientific Reports*, vol. 7, no. 1, article no. 4236, 2017.

[12] S. Pérez-Landero, S. Sandoval-Motta, C. Martínez-Anaya et al., "Complex regulation of Hsf1-Skn7 activities by the catalytic subunits of PKA in Saccharomyces cerevisiae: Experimental and computational evidences," *BMC Systems Biology*, vol. 9, no. 1, article no. 42, 2015.

[13] S. Barbosa, B. Niebel, S. Wolf, K. Mauch, and R. Takors, "A guide to gene regulatory network inference for obtaining predictive solutions: Underlying assumptions and fundamental biological and data constraints," *BioSystems*, vol. 174, pp. 37–48, 2018.

[14] B. A. McGregor, S. Eid, A. E. Rumora et al., "Conserved transcriptional signatures in human and murine diabetic peripheral neuropathy," *Scientific Reports*, vol. 8, no. 1, 2018.

[15] S. Barman and Y.-K. Kwon, "A novel mutual information-based Boolean network inference method from time-series gene expression data," *PLoS ONE*, vol. 12, no. 2, Article ID e0171097, 2017.

[16] Z.-P. Liu, C. Wu, H. Miao, and H. Wu, "RegNetwork: An integrated database of transcriptional and post-transcriptional regulatory networks in human and mouse," *Database*, vol. 2015, pp. 1–12, 2015.

[17] A. Krämer, J. Green, J. Pollard, and S. Tugendreich, "Causal analysis approaches in ingenuity pathway analysis," *Bioinformatics*, vol. 30, no. 4, pp. 523–530, 2014.

[18] N. A. O'Leary, M. W. Wright, J. R. Brister et al., "Reference sequence (RefSeq) database at NCBI: Current status, taxonomic expansion, and functional annotation," *Nucleic Acids Research*, vol. 44, no. 1, pp. D733–D745, 2016.

[19] D. Worku, F. Jouhra, G. W. Jiang, N. Patani, R. F. Newbold, and K. Mokbel, "Evidence of a tumour suppressive function of E2F1 gene in human breast cancer," *Anticancer Reseach*, vol. 28, no. 4 B, pp. 2135–2139, 2008.

[20] H. Tovar, R. García-Herrera, J. Espinal-Enríquez, and E. Hernández-Lemus, "Transcriptional master regulator analysis in breast cancer genetic networks," *Computational Biology and Chemistry*, vol. 59, pp. 67–77, 2015.

[21] J. Espinal-Enriquez, R. A. Meja-Pedroza, and E. Hernndez-Lemus, "A Boolean network model for invasive thyroid carcinoma," in *Proceedings of the Artificial Life Conference 2016*, pp. 570–577, Cancun, Mexico, July 2016.

[22] J. Wang, C. Li, and C. Xia, "Improved centrality indicators to characterize the nodal spreading capability in complex networks," *Applied Mathematics and Computation*, vol. 334, pp. 388–400, 2018.

[23] C. Xia, X. Li, Z. Wang, and M. Perc, "Doubly effects of information sharing on interdependent network reciprocity," *New Journal of Physics*, vol. 20, no. 7, Article ID 075005, 2018.

[24] C. Chen, Y. Hu, and L. Li, "NRP1 is targeted by miR-130a and miR-130b, and is associated with multidrug resistance in epithelial ovarian cancer based on integrated gene network analysis," *Molecular Medicine Reports*, vol. 13, no. 1, pp. 188–196, 2016.

[25] M. Leone, A. Pagnani, G. Parisi, and O. Zagordi, "Finite size corrections to random Boolean networks," *Journal of Statistical Mechanics: Theory and Experiment*, no. 12, Article ID P12012, 2006.

[26] B. Derrida and Y. Pomeau, "Random networks of automata: A simple annealed approximation," *EPL (Europhysics Letters)*, vol. 1, no. 2, pp. 45–49, 1986.

[27] U. Bastolla and G. Parisi, "The modular structure of Kauffman networks," *Physica D: Nonlinear Phenomena*, vol. 115, no. 3-4, pp. 219–233, 1998.

[28] M. W. Hirsch, R. L. Devaney, and S. Smale, *Differential Equations, Dynamical Systems, and Linear Algebra*, vol. 6, Academic Press, New York, NY, USA, 1974.

Hindawi
Complexity
Volume 2019, Article ID 2108014, 13 pages
https://doi.org/10.1155/2019/2108014

Research Article

Binomial Representation of Cryptographic Binary Sequences and Its Relation to Cellular Automata

Sara D. Cardell [1] **and Amparo Fúster-Sabater** [2]

[1] *IMECC, University of Campinas (UNICAMP), Brazil*
[2] *Institute of Physical and Information Technologies, C.S.I.C., Madrid, Spain*

Correspondence should be addressed to Sara D. Cardell; sdcardell@ime.unicamp.br

Received 23 December 2018; Accepted 13 February 2019; Published 24 March 2019

Academic Editor: Jose C. Valverde

The binomial sequences are binary sequences that correspond to the diagonals of the binary Sierpinski's triangle. They have fancy properties such that all the sequences with period equal to a power of 2 can be represented as the sum of a finite set of binomial sequences. Other structural properties of these sequences (period, linear complexity, construction rules, or relations among the different binomial sequences) have been analyzed in detail. Furthermore, this work enhances the close relation between the binomial sequences and a kind of Boolean networks, known as linear cellular automata. In this sense, the binomial sequences exhibit the same behavior as that of particular Boolean networks. Consequently, the binomial sequences can be considered as primary tools for generating other more complex Boolean networks with applications in communication systems and cryptography.

1. Introduction

Pseudorandom binary sequences are simple successions of bits with applications in fields so different as spread-spectrum communications, circuit testing, error-correcting codes, numerical simulations, or cryptography (stream cipher). Most generators producing such sequences are based on Boolean functions and Linear Feedback Shift Registers (LFSRs) [1]. Desirable characteristics for pseudorandom binary sequences are long period, good statistical properties or large linear complexity. Different LFSR-based sequence generators can be found in the literature [2, Chapter 5]. In most of them, the output sequence is a binary sequence generated as the image of a nonlinear Boolean function in the shift register binary stages.

On the other hand, the binomial sequences are a family of binary sequences whose terms are binomial numbers reduced modulo 2. More precisely, the binomial sequences correspond to the diagonals of the Sierpinski's triangle modulo 2. In this way, the binomial sequences exhibit many attractive properties that can be very useful in the analysis and generation of cryptographic sequences. In this work, it is shown that every binary sequence with period 2^L, L being a positive integer, can be written as a bit-wise XOR of binomial sequences.

Since many of the cryptographic sequences have period 2^L [3–6], then the binomial sequences can be considered as a fundamental tool to analyze the structural properties of all these classes of sequences. In addition, it can be checked that the behavior of some binomial sequence combinations is the same as that of a kind of Boolean networks (namely, one-dimensional cellular automata). In fact, cellular automata with two-state cells is a special kind of Boolean network where all the nodes use the same function and the links are all arranged in a regular bounded integer lattice structure. Boolean networks have attracted great attention in many different areas such as bioinformatics [7], computational processes [8], graph dynamical systems [9], and parallel discrete dynamical systems [10, 11]. This paper shows the subtle relation between binomial sequences and cellular automata. In brief, the binomial sequences and the linear cellular automata make visible the linearity inherent to many cryptographic generators paradoxically designed as strong nonlinear generators.

The paper is organized as follows: In Section 2, we introduce the basic concepts and definitions needed for the rest of this work. Section 3 studies the characterization and main properties of the binomial sequences. In Section 4, the relation between binomial sequences and linear cellular automata is analyzed. A simple method of recovering the binomial representation of a sequence is developed in Section 5 with an example. Finally, conclusions in Section 6 end the paper.

2. Preliminaries

In this section, we present some basic concepts about sequences that we need to know before introducing the main results.

2.1. Binary Sequences. Let $\mathbb{F}_2$ be the Galois field of two elements. We say $\{a_n\} = \{a_0, a_1, a_2 \ldots\}$ is a binary sequence if its terms $a_n \in \mathbb{F}_2$, for $n = 0, 1, 2, \ldots$. The sequence $\{a_n\}$ is periodic if and only if there exists an integer T such that $a_{n+T} = a_n$, for all $n \geq 0$. In the sequel, all the sequences considered will be binary sequences and the XOR operation among sequences will be denoted by + instead of the symbol $\oplus$.

Let r be a positive integer, and let $d_1, d_2, d_3, \ldots, d_r$ be constant coefficients with $d_i \in \mathbb{F}_2$. A binary sequence $\{a_n\}$ satisfying the relation

$$a_{n+r} = d_r a_n + d_{r-1} a_{n+1} + \cdots + d_3 a_{n+r-3} + d_2 a_{n+r-2} + d_1 a_{n+r-1}, \quad n \geq 0, \tag{1}$$

is called a (r-th order) linear recurring sequence in $\mathbb{F}_2$. The terms $\{a_0, a_1, \ldots, a_{r-1}\}$ are referred to as the initial values (or initial state) and determine the rest of the sequence uniquely. A relation of the form given by (1) is called a (r-th order) linear recurrence relationship.

The monic polynomial

$$p(x) = d_r + d_{r-1} x + \cdots + d_3 x^{r-3} + d_2 x^{r-2} + d_1 x^{r-1} + x^r \in \mathbb{F}_2[x] \tag{2}$$

is called the characteristic polynomial of the linear recurring sequence and $\{a_n\}$ is said to be generated by $p(x)$.

The generation of linear recurring sequences can be implemented on LFSRs [1]. These structures handle information in the form of binary elements and they are based on shifts and linear feedback. In fact, an LFSR is an electronic device with r memory cells (stages) with binary contents. At each time instant, each element is shifted to the adjacent stage and a new element is computed via a linear feedback to fill the empty stage (see Figure 1). If the characteristic polynomial of the linear recurring sequence is primitive [1], then the LFSR is a maximal-length LFSR and its output sequence, the so-called PN-sequence, has period $T = 2^r - 1$.

The linear complexity, LC, of a sequence $\{a_n\}$ is defined as the length of the shortest LFSR that generates such a sequence or, equivalently, as the lowest order linear recurrence relationship that generates such a sequence.

In cryptographic terms, the linear complexity must be as large as possible. The recommended value is approximately half the period $LC \simeq T/2$.

Let E be the shifting operator that acts on the terms of a sequence $\{a_n\}$; that is,

$$E^k a_n = a_{n+k}, \quad \text{for all integer } k \geq 0. \tag{3}$$

The linear recurrence relationship given in (1) can be written in terms of the operator E as a linear difference equation:

$$\left(E^r + \sum_{j=1}^{r} d_j E^{r-j}\right) a_n = 0, \quad \text{for } n \geq 0. \tag{4}$$

If the characteristic polynomial $p(x)$ is a primitive polynomial of degree r and $\alpha \in \mathbb{F}_{2^r}$ is one of its roots, then $\alpha, \alpha^2, \alpha^{2^2}, \ldots, \alpha^{2^{r-1}}$ are the r roots of such a polynomial. In this case, the binary solutions of (4) are a linear combination of the r roots of the form

$$a_n = \sum_{j=0}^{r-1} c_i^{2^j} \alpha^{2^j n}, \tag{5}$$

that is, a_n is the n-th term of a PN-sequence with characteristic polynomial $p(x)$ and whose initial values are determined by the coefficient $c_i \in \mathbb{F}_{2^r}$.

Next, let us consider a bit more complex difference equation of the form

$$\left(E^r + \sum_{j=1}^{r} d_j E^{r-j}\right)^m z_n = 0, \quad \text{for } n \geq 0, \tag{6}$$

whose characteristic polynomial is $p_m(x) = p(x)^m = (x^r + \sum_{j=1}^{r} d_j x^{r-j})^m$, m being a positive integer. Now, the roots of $p_m(x)$ are the same as those of $p(x)$ but with multiplicity m. Therefore, the binary solutions of (6) are given by

$$z_n = \sum_{i=0}^{m-1} \left[\binom{n}{i} \sum_{j=0}^{r-1} c_i^{2^j} \alpha^{2^j n} \right], \tag{7}$$

where the coefficients $c_i \in \mathbb{F}_{2^r}$ and $\binom{n}{i}$ are binomial coefficients reduced modulo 2 (see [12]). Since

$$\sum_{j=0}^{r-1} c_i^{2^j} \alpha^{2^j n} \tag{8}$$

is the n-th term of a PN-sequence with characteristic polynomial $p(x)$ and initial values determined by c_i, then z_n is the sum of m terms of a unique PN-sequence starting at different points and where each one of these m terms is weighted by a binary binomial coefficient $\binom{n}{i}$.

Table 1: Binomial sequences, first eight terms, period, and linear complexity.

Binomial coeff.	Binomial sequences	Period	Linear complexity
$\binom{n}{0}$	1 1 1 1 1 1 1 1	$T_0 = 1$	$LC_0 = 1$
$\binom{n}{1}$	0 1 0 1 0 1 0 1	$T_1 = 2$	$LC_1 = 2$
$\binom{n}{2}$	0 0 1 1 0 0 1 1	$T_2 = 4$	$LC_2 = 3$
$\binom{n}{3}$	0 0 0 1 0 0 0 1	$T_3 = 4$	$LC_3 = 4$
$\binom{n}{4}$	0 0 0 0 1 1 1 1	$T_4 = 8$	$LC_4 = 5$
$\binom{n}{5}$	0 0 0 0 0 1 0 1	$T_5 = 8$	$LC_5 = 6$
$\binom{n}{6}$	0 0 0 0 0 0 1 1	$T_6 = 8$	$LC_6 = 7$
$\binom{n}{7}$	0 0 0 0 0 0 0 1	$T_7 = 8$	$LC_7 = 8$

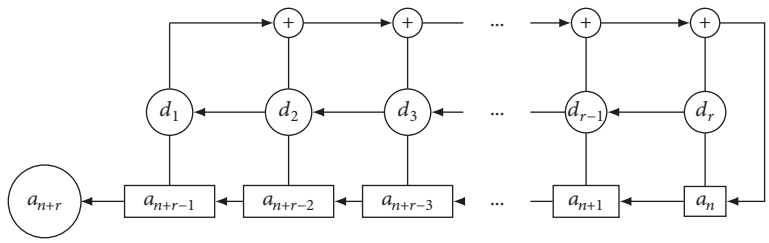


Figure 1: LFSR of length r.

3. Binomial Sequences

Previous to the introduction of the binomial sequence concept, let us consider some general features of the binomial coefficients.

The binomial coefficient $\binom{n}{i}$ is the coefficient of the power x^i in the polynomial expansion of $(1 + x)^n$. For every positive integer n, it is a well-known fact that $\binom{n}{0} = 1$ and $\binom{n}{i} = 0$ for $i > n$. Moreover, it is worth noticing that if we arrange these binomial coefficients into rows for successive values of $n = 0, 1, 2, \ldots$, then the generated structure is the Pascal's triangle (see Figure 2(a)). The most-left diagonal is the identically 1 sequence, the next diagonal is the sequence of natural numbers $\{1, 2, 3, \ldots\}$, the next one is the sequence of triangular numbers $\{1, 3, 6, 10, \ldots\}$, etc. Other fascinating sequences (tetrahedral numbers, pentatope numbers, hexagonal numbers, Fibonacci sequence, etc.) can be found in the diagonals of this triangle. On the other hand, if we color the odd numbers of the Pascal's triangle and shade the even numbers, then we get the Sierpinski's triangle (see Figure 2(b)).

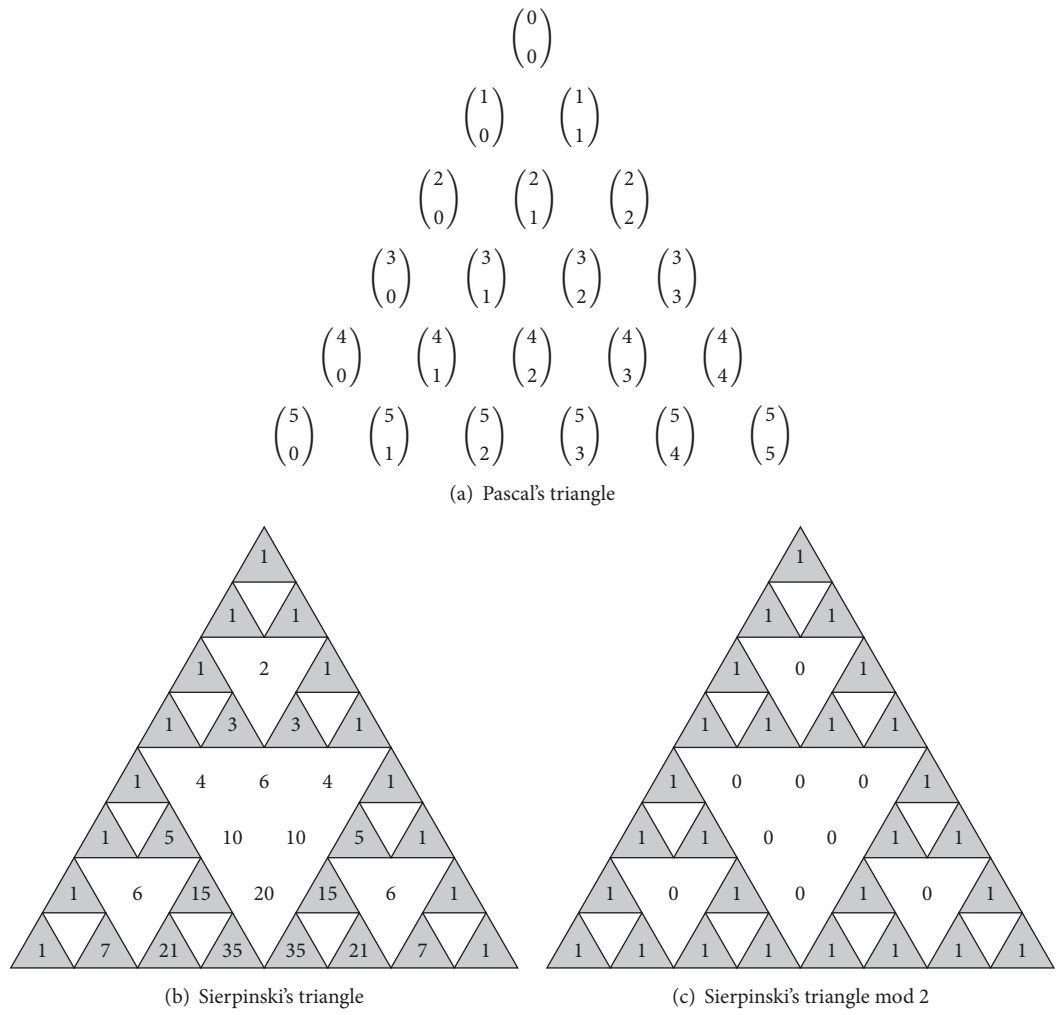
The binomial coefficients reduced modulo 2 allow us to introduce the concept of binomial sequence.

Definition 1. Given a fixed integer $k \geq 0$, the sequence $\{b_n^k\}_{n \geq 0}$ given by

$$b_n^k = \begin{cases} 0 & \text{if } n < k \\ \binom{n}{k} \bmod 2 & \text{if } n \geq k \end{cases} \tag{9}$$

is known as the *binary k-th binomial sequence*.

Table 1 shows the binomial sequences and their corresponding periods and linear complexities, denoted by T_i and LC_i, respectively, for the first 8 binomial coefficients $\binom{n}{i}$, $i = 0, 1, \ldots, 7$; see [12]. The linear complexities of the binomial sequences are defined in Theorem 13 (Section 4). Recall that the successive binomial sequences correspond to shifted versions of the successive diagonals in the Sierpinski's triangle reduced modulo 2 (see Figure 2(c)).



(a) Pascal's triangle

(b) Sierpinski's triangle

(c) Sierpinski's triangle mod 2

Figure 2: Binomial coefficients arranged as triangles.

Next, the relation between binomial sequences and binary sequences with period which is a power of 2 is defined in the following result.

Theorem 2. *Let $\{z_n\}$ be a binary sequence with period $T = 2^L$, L being a positive integer. Then, every binary sequence $\{z_n\}$ can be written as a linear combination of binomial sequences.*

Proof. Since the period of $\{z_n\}$ is a power of 2, then the next equation holds:

$$\left(E^{2^L} + 1\right) z_n = (E+1)^{2^L} z_n = 0, \tag{10}$$

which is a simplified version of (6) with $m = 2^L$ and the characteristic polynomial $p_m(x) = p(x)^m = (x+1)^m$. Therefore, its binary solutions are given by (7), which has now the following simplified form:

$$z_n = \binom{n}{0} c_0 + \binom{n}{1} c_1 + \cdots + \binom{n}{T-1} c_{T-1} \quad \text{for } n \geq 0, \tag{11}$$

where 1 is the unique root of the polynomial $(x+1)^T$ with multiplicity $T = 2^L$, the coefficients $c_i \in \mathbb{F}_2$ and $\binom{n}{i}$ are binomial coefficients modulo 2. When n takes successive values $n = 0, 1, 2, \ldots$, then each binomial coefficient modulo 2 defines a different binomial sequence. Thus, the sequence $\{z_n\}$ is just the bit-wise XOR of such binomial sequences weighted by binary coefficients c_i. □

Different choices of c_i will produce different sequences $\{z_n\}$ with distinct characteristics and properties, but all of them with period 2^l, $0 \leq l \leq L$.

4. Properties of the Binomial Sequences

From now on, we denote the k-th binomial sequence $\{b_n^k\}_{n\geq 0}$ as $\left\{\binom{n}{k}\right\}_{n\geq 0}$ or simply $\left\{\binom{n}{k}\right\}$, while $\binom{n}{k}$ denotes a binomial coefficient.

In this section we study the properties of this family of binomial sequences.

Next result shows that the binomial sequences can be obtained one from another.

Proposition 3. *Given the sequence* $\{\binom{n}{2^L+k}\}$, *with* $0 \le k < 2^L$, *we have that*

(a) *the sequence has period* $T = 2^{L+1}$;

(b) *the first period of the sequence has the following structure:*

$$\left\{\binom{n}{2^L+k}\right\}_{0\le n<2^{L+1}} = \begin{cases} 0 & \text{if } 0 \le n < 2^L + k, \\ \binom{n}{k} & \text{if } 2^L + k \le n < 2^{L+1}. \end{cases} \tag{12}$$

Proof. (b) We consider the first 2^{L+1} bits of the sequence $\{\binom{n}{2^L+k}\}$.

We know that $\binom{n}{2^L+k} = 0$ when $n < 2^L + k$. Then, the first $2^L + k$ bits are 0s; in particular, this means that the first 2^L elements of the sequence are zero.

If $n \ge 2^L + k$, then n is of the form $n = 2^L + k + t$, for $0 \le t < 2^L - k$. We want to prove that the other 2^L bits are the first 2^L bits of $\{\binom{n}{k}\}$. This idea is illustrated in Figure 3.

In order to prove that the other 2^L bits are the first 2^L bits of $\{\binom{n}{k}\}$, it is enough to prove that $\binom{2^L+k+t}{2^L+k} \equiv \binom{k+t}{k} \pmod 2$.

Thus, we compute both binomial coefficients

$$\begin{aligned} \binom{2^L+k+t}{2^L+k} &= \frac{(2^L+k+t)!}{(2^L+k)!t!} \\ &= \frac{(2^L+k+t)\cdots(2^L+k+1)}{t!}, \\ \binom{k+t}{k} &= \frac{(k+t)\cdots(k+1)}{t!} \end{aligned} \tag{13}$$

Let 2^{p_i} be the maximum power of 2 in the prime factorization of $k + i$, with $0 < i \le t$ and q_i the odd number such that $k + i = 2^{p_i} q_i$. Notice that when $k + i$ is odd, then $k + i = q_i$ and $p_i = 0$.

Then, we have

$$\begin{aligned} &\binom{2^L+k+t}{2^L+k} \\ &= \frac{2^{p_t+p_{t-1}+\cdots+p_1}\left(2^{L-p_t}+q_t\right)\left(2^{L-p_{t-1}}+q_{t-1}\right)\cdots\left(2^{L-p_1}+q_1\right)}{t!}, \end{aligned} \tag{14}$$

$$\binom{k+t}{k} = \frac{2^{p_t+p_{t-1}+\cdots+p_1} q_t \cdot q_{t-1}\cdots q_1}{t!} \tag{15}$$

Since $k + i < 2^L$, then $2^{p_i} q_i < 2^L$ and, as a consequence, $p_i < L$. Now, the inequality $L - p_i > 0$ implies that $2^{L-p_i} + q_i$ is always an odd number. Finally, since both (14) and (15) have the same denominator, then they exhibit the same powers of two. Thus, (14) is odd (even) iff (15) is odd (even) and the previous congruence holds.

(a) It is enough to prove now that

$$\binom{2^{L+m}+t}{2^L+k} \equiv \binom{2^{L+m+1}+t}{2^L+k} \bmod 2 \tag{16}$$

We consider both binomial coefficients:

$$\begin{aligned} &\binom{2^{L+m}+t}{2^L+k} \\ &= \frac{\left(2^{L+m}+t\right)\cdot\left(2^{L+m}+t-1\right)\cdots\left(2^{L+m}+t-2^L-k+1\right)}{(2^L+k)!} \\ &\binom{2^{L+m+1}+t}{2^L+k} \\ &= \frac{\left(2^{L+m+1}+t\right)\cdot\left(2^{L+m+1}+t-1\right)\cdots\left(2^{L+m+1}+t-2^L-k+1\right)}{(2^L+k)!} \end{aligned} \tag{17}$$

Consider 2^{p_i} the maximum power of 2 in the prime factorization of i, with $t - 2^L - k + 1 \le i \le t$ and q_i the odd number such that $2^{p_i} \cdot q_i = i$. Notice that when i is odd, then $p_i = 0$ and $i = q_i$. With this new notation, we have that

$$\binom{2^{L+m}+t}{2^L+k} = \frac{2^{p_t\cdot p_{t-1}\cdots p_{t-2^L-k+1}}\left(2^{L+m-p_t}+q_t\right)\cdot\left(2^{L+m-p_{t-1}}+q_{t-1}\right)\cdots\left(2^{L+m-p_{t-2^L-k+1}}+q_{t-2^L-k+1}\right)}{(2^L+k)!} \tag{18}$$

$$\binom{2^{L+m+1}+t}{2^L+k} = \frac{2^{p_t\cdot p_{t-1}\cdots p_{t-2^L-k+1}} q_t \cdot q_{t-1}\cdots q_{t-2^L-k+1}}{(2^L+k)!} \tag{19}$$

Note that $L + m - p_i > 0$ and then $2^{L+m-p_i} + q_i$ is always an odd number. Now, since both expressions (18) and (19) have the same denominator and the same powers of two in the prime factorization of the numerator, we know that (18) is odd (even) iff (19) is odd (even).

We have proven that $\binom{n}{2^L+k} = \binom{n+2^{L+1}}{2^L+k}$, for $n \ge 0$. Then we know that the period divides 2^{L+1}. Since the first 2^L bits of the sequence are 0s (item b), then the period must be $T = 2^{L+1}$. □

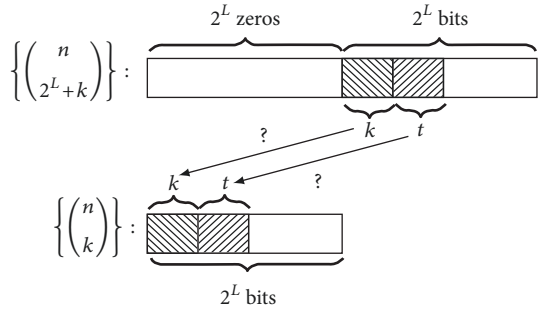


FIGURE 3: Structure of the binomial sequences $\{\binom{n}{k}\}$ and $\{\binom{n}{2^L+k}\}$.

In Figure 8, we can see the structure of the first 32 binomial sequences. It is easy to observe the pattern and the periods mentioned in Proposition 3.

It is worth noticing that the binomial sequences match exactly with the diagonals of the binary Sierpinski's triangle (see Figure 4) but starting in a different bit (shifted versions of such diagonals). For example, the encircled sequence in Figure 4 corresponds to the shifted binomial sequence $\{\binom{n}{4}\}$.

We know that the sequence $\{\binom{n}{k}\}$ is a solution of the difference equation of the form (10). Therefore, every sequence of period 2^L can be obtained by XORing diagonals of the binary Sierpinski's triangle.

Corollary 4. *The sequences* $\{\binom{n}{2^L}\}$ $(L = 0, 1, 2, \ldots)$ *have period* $T = 2^{L+1}$ *and the following structure:*

$$\left\{\binom{n}{2^L}\right\}_{0\le n<2^{L+1}} = \begin{cases} 0 & \text{if } 0 \le n < 2^L, \\ 1 & \text{if } 2^L \le n < 2^{L+1}. \end{cases} \tag{20}$$

Corollary 5. *The sequences* $\{\binom{n}{2^L}\}$ $(L = 0, 1, 2, \ldots)$ *are balanced; that is to say, they contain the same number of 1s and 0s.*

Remark 6. (a) The sequences of the form $\{\binom{n}{2^L}\}$ have the following structure:

$$\underbrace{0\ 0\ \ldots 0}_{2^L \text{ zeros}}\underbrace{1\ 1\ \ldots\ 1}_{2^L \text{ ones}} \tag{21}$$

(b) The sequences of the form $\{\binom{n}{2^L+k}\}$ have the following structure:

$$\underbrace{0\ 0\ \ldots 0}_{2^L \text{ zeros}}\underbrace{\ldots\ldots\ldots\ldots}_{\substack{2^L \text{ first terms} \\ \text{of } \{\binom{n}{k}\}}} \tag{22}$$

According to Theorem 2, a binary sequence of period power of 2 is the bit-wise XOR of binomial sequences. Therefore, we introduce the following definition.

Definition 7. The set of binomial sequences necessary to obtain a binary sequence of period power of 2 is called the *binomial representation* of such a sequence.

The binomial representation of a sequence is of the form $\sum_{i=0}^{M} c_i \{\binom{n}{i}\}$, with $c_i \in \mathbb{F}_2$ and M an integer such that $M \ge 0$.

Since our sequences are periodic, they can start in different points. Next we see that, depending on the starting point, the binomial representations of the same sequence will be different.

Lemma 8. *Given two positive integers* n *and* t *with* $n > t$, *we have the following:*

$$\binom{n}{t} + \binom{n}{t-1} = \binom{n+1}{t} \tag{23}$$

Proof.

$$\begin{aligned} \binom{n}{t} + \binom{n}{t-1} &= \frac{n!}{t!\,(n-t)!} + \frac{n!}{(t-1)!\,(n-t+1)!} \\ &= \frac{(n-t+1)\cdot n! + t\cdot n!}{t!\,(n-t+1)!} \\ &= \frac{(n+1)!}{t!\,(n+1-t)!} = \binom{n+1}{t} \end{aligned} \tag{24}$$

□

Lemma 9. *Given the binomial sequence* $\{\binom{n}{t}\}$, $t \ge 1$, *if we shift cyclically such a sequence one bit to the left, then we obtain the sequence* $\{\binom{n}{t} + \binom{n}{t-1}\}$. *If* $t = 0$, *the sequence remains the same (in this case the sequence is the identically 1 sequence).*

Proof. According to the construction rule for binomial sequences given in Definition 1, the sequences $\{\binom{n}{t}\}$ and $\{\binom{n+1}{t}\}$ are the same but starting in different points.

Now, according to Lemma 8, we know that $\binom{n}{t} + \binom{n}{t-1} = \binom{n+1}{t}$, then the sequence $\{\binom{n+1}{t}\}$ equals the sequence $\{\binom{n}{t} + \binom{n}{t-1}\}$. Therefore, the sequences $\{\binom{n}{t}\}$ and $\{\binom{n}{t} + \binom{n}{t-1}\}$ are the same but starting in different bits. □

Example 10. Consider the following sequences:

$$\begin{aligned} \left\{\binom{n}{2}\right\} &: \quad \{0\ \mathbf{0}\ 1\ 1\ 0\ 0\ 1\ 1\ \ldots\} \\ \left\{\binom{n}{2} + \binom{n}{1}\right\} &: \quad \{\mathbf{0}\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ \ldots\} \end{aligned} \tag{25}$$

Both sequences $\{\binom{n}{2}\}$ and $\{\binom{n}{2} + \binom{n}{1}\}$ are the same, but starting in different positions. We can check that the starting point of the sequence $\{\binom{n}{2} + \binom{n}{1}\}$ (bit in bold) is the second bit of sequence $\{\binom{n}{2}\}$.

In order to prove the linear complexity of the binomial sequences, we need to introduce the following results.

Proposition 11. *Given the binomial sequence* $\{\binom{n}{t}\}$, *with a fixed* $t \ge 1$, *the sequence represented by* $\{\binom{n}{t} + \binom{n}{t+1}\}$ *can be also represented by* $\{\binom{n}{t-1}\}$. *If* $t = 0$, *the sequence* $\{\binom{n}{t} + \binom{n}{t+1}\}$ *is the identically zero sequence.*

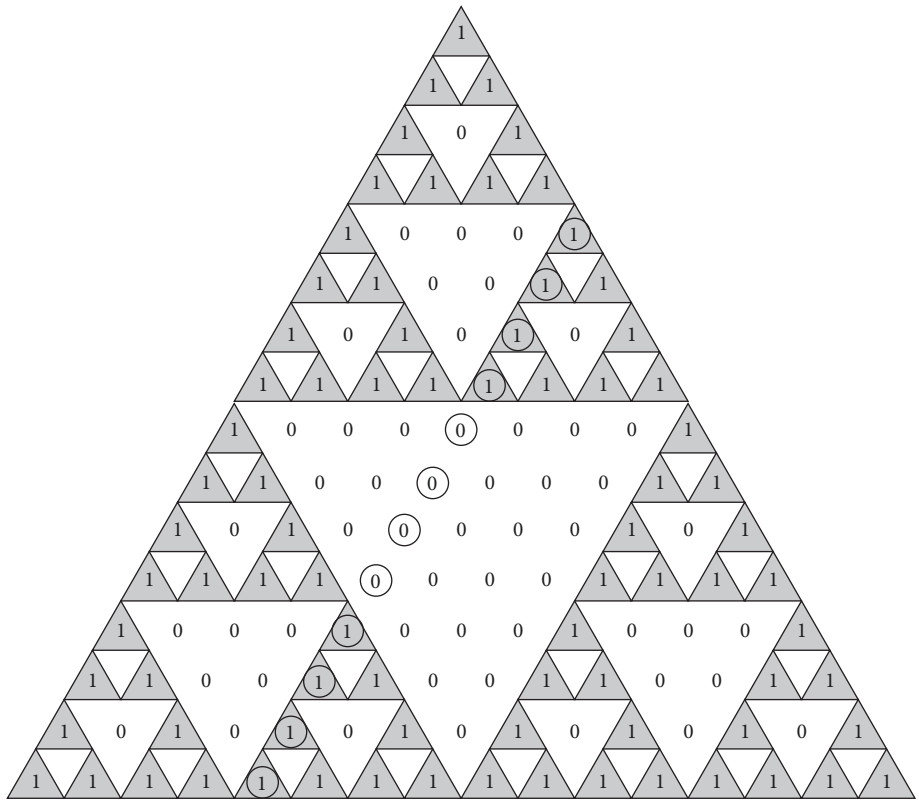


Figure 4: Binary Sierpinski's triangle where the circled bits correspond to the sequence $\{\binom{n}{4}\}$.

Proof. According to Lemma 8, we know that $\binom{n}{t} + \binom{n}{t-1} = \binom{n+1}{t}$. Since we are working over the binary field, we have that the sequence $\{\binom{n}{t} + \binom{n}{t+1}\}$ can be also represented by $\{\binom{n}{t-1}\}$. □

Theorem 12 ([13], Theorem 1). *Let $\{s_i\}_{i\geq 0}$ be a binary sequence whose characteristic polynomial is $(1+x)^n$. Then, the characteristic polynomial of the sequence $\{u_i\}_{i\geq 0}$, where $u_i = s_i + s_{i+1}$, is $(1+x)^{n-1}$.*

Now, we are ready to study the linear complexity of the binomial sequences.

Theorem 13. *The linear complexity of the sequence $\{\binom{n}{k}\}$ is $k+1$.*

Proof. We prove this result by induction.

For $k = 0$, the sequence $\{\binom{n}{0}\} = \{1\ 1\ 1\ 1\ 1\ \ldots\}$ has $LC_0 = 1$ and the characteristic polynomial is $(1+x)$.

For $k = 1$, the sequence $\{\binom{n}{1}\} = \{0\ 1\ 0\ 1\ \ldots\}$ represented by $\{\binom{n}{1}\}$ has $LC_1 = 2$ and the characteristic polynomial is $(1+x)^2$.

Let us suppose that the sequence $\{\binom{n}{k}\}$ has $LC_k = k+1$ and the characteristic polynomial is $(1+x)^{k+1}$.

According to Proposition 11, we have that $\{\binom{n}{k}\} = \{\binom{n}{k+1} + \binom{n+1}{k+1}\}$. Now, according to Theorem 12, the characteristic polynomial of $\{\binom{n}{k+1}\}$ is $(1+x)^{k+2}$ and, thus, $LC_{k+1} = k+2$. □

As a consequence of the previous theorem, we have the following result.

Corollary 14. *Given a sequence with binomial representation $\sum_{k=1}^{t}\{\binom{n}{i_k}\}$, where $i_1 < i_2 < \cdots < i_t$ are integer indices, then the linear complexity of such a sequence is $i_t + 1$.*

As a consequence of Lemma 9, we obtain the following result.

Theorem 15. *Let $\sum_{i=0}^{LC-1} c_i\{\binom{n}{i}\}$ with $c_i \in \mathbb{F}_2$ be the binomial representation of a sequence. If we shift cyclically such a sequence one bit to the left, then its binomial representation is*

$$\sum_{i=1}^{LC-1} c_i\left[\left\{\binom{n}{i} + \binom{n}{i-1}\right\}\right] + c_0\left\{\binom{n}{0}\right\}. \tag{26}$$

Corollary 16. *Binary sequences with period 2^L have 2^L different binomial representations.*

Proof. Since the period of the sequence is 2^L, we can perform 2^L left shifts before getting the same starting point of the sequence. Therefore, we can obtain 2^L different binomial representations. □

Example 17. Consider the sequence with binomial representation $\{\binom{n}{3} + \binom{n}{1}\}$. In Figure 5, we can see graphically the

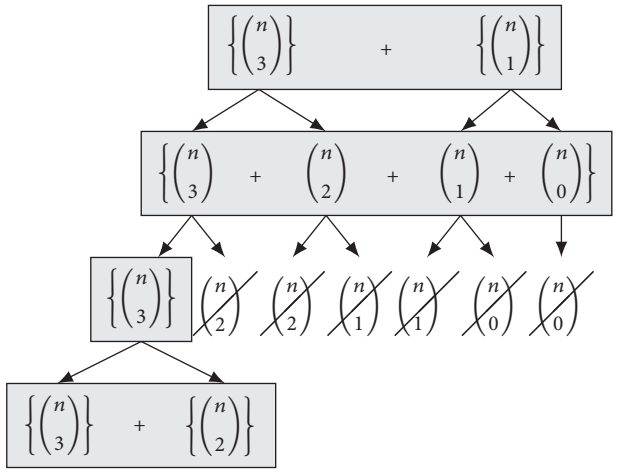


Figure 5: Binomial representations of the 4 shifted versions of the sequence $\{\binom{n}{3} + \binom{n}{1}\}$.

Table 2: Rules 102 and 60.

(a) *Rule 102:* $x_i^{t+1} = x_i^t + x_{i+1}^t$

111	110	101	100	011	010	001	000
0	1	1	0	0	1	1	0

(b) *Rule 60:* $x_i^{t+1} = x_{i-1}^t + x_i^t$

111	110	101	100	011	010	001	000
0	0	1	1	1	1	0	0

method followed to obtain the different binomial representations of this sequence. From one representation and via Theorem 15, we obtain the next representation corresponding to the same sequence left-shifted one bit. Finally, we have 4 different representations (the ones in bold contained in the grey boxes) including the initial one:

$$\begin{gathered}\left\{\binom{n}{3} + \binom{n}{1}\right\},\\ \left\{\binom{n}{3} + \binom{n}{2} + \binom{n}{1} + \binom{n}{0}\right\},\\ \left\{\binom{n}{3}\right\},\\ \left\{\binom{n}{3} + \binom{n}{2}\right\}.\end{gathered} \tag{27}$$

Since the period of this sequence is 4, we can obtain 4 different binary representations. Furthermore, one can observe that after four steps we obtain again the initial representation $\{\binom{n}{3} + \binom{n}{1}\}$.

Now, consider again Figure 4. We know that the binomial sequence $\{\binom{n}{4}\}$ showed in the binary Sierpinski's triangle starts in a different bit compared with the sequence $\{\binom{n}{4}\}$ given in Table 1. In particular, in the binary Sierpinski's triangle the sequences start in the first nonzero bit; thus their binomial representations are different. For instance, consider Table 3 where we can observe the different representations of a unique binomial sequence $\{\binom{n}{15}\}$. Each row represents the coefficients $\{c_0, c_1, \ldots, c_{15}\}$ of each different binomial representation. The binomial representation of the sequence $\{\binom{n}{15}\}$ in the binary Sierpinski's triangle is the last row of Table 3: $\sum_{i=0}^{15} c_i \{\binom{n}{i}\}$, with $c_i = 1$, for $i = 0, 1, \ldots, 15$.

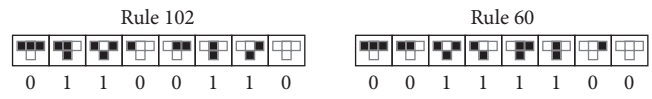


Figure 6: Rules 102 and 60 depicted in Wolfram's notation.

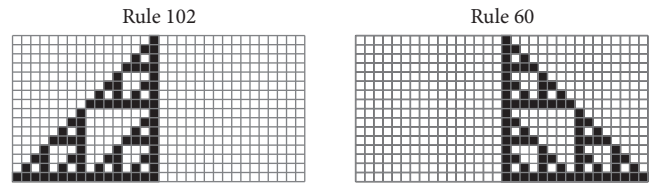


Figure 7: CA-images generated with rules 102 and 60.

5. Cellular Automata

Cellular automata (CA) are discrete structures composed of a finite number of cells whose content is updated according to a *rule* or function with k variables [14]. The state of the cell in position i at time $t + 1$, notated x_i^{t+1}, depends on the state of the k neighbour cells at time t. If these rules are composed exclusively of XOR operations, then the CA are *linear*. Here, the CA we consider are *regular* (every cell follows the same rule), *cyclic* (extreme cells are adjacent), and one-dimensional. For $k = 3$, rules 102 and 60 are given in Table 2.

The number 01100110 (00111100) is the binary representation of the decimal number 102 (60). In Figure 6, these rules are depicted according to Wolfram terminology [15]: a white square represents the digit 0 and a black square represents the digit 1.

Consider again Table 3. If we color the 1s, the general structure of the set of characterizations is the same as that one of the CA-image generated by rule 102 after having applied 15 iterations to the one-dimensional cellular automata (see Figure 7). In general, due to the observed form of the binomial sequences (see Figure 8 and Proposition 3), it can be assured that the complete set of binomial representations of $\{\binom{n}{2^L-1}\}$ coincides with the 102-CA of length and 2^L and initial state $\{0\ 0\ \ldots\ 0\ 1\}$. This is due to the fact that the recursive method to obtain the different binomial representations of a sequence matches with the generation rule of 102-CA (depicted in Table 2).

As a consequence, we can introduce the following result.

Theorem 18. *Consider a sequence* $\{u_j\}_{j\geq 0}$ *with binomial representation* $\sum_{i=0}^{LC-1} c_i \{\binom{n}{i}\}$. *If we put this sequence in the*

Table 3: Binomial representations of the 16 shifted versions of $\{\binom{n}{15}\}$.

$\binom{n}{0}$	$\binom{n}{1}$	$\binom{n}{2}$	$\binom{n}{3}$	$\binom{n}{4}$	$\binom{n}{5}$	$\binom{n}{6}$	$\binom{n}{7}$	$\binom{n}{8}$	$\binom{n}{9}$	$\binom{n}{10}$	$\binom{n}{11}$	$\binom{n}{12}$	$\binom{n}{13}$	$\binom{n}{14}$	$\binom{n}{15}$
c_0	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	1
0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1
0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1
0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	1
0	0	0	0	0	1	0	1	0	0	0	0	0	1	0	1
0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1
0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

leftmost column of a 102-CA (rightmost column of a 60-CA), then the binomial representation of the next sequence $\{u_j + u_{j+1}\}_{j\geq 0}$ *in the CA is*

$$\sum_{i=0}^{LC-2} c_{i+1} \left\{\binom{n}{i}\right\}. \tag{28}$$

Proof. Let us denote the binomial sequence $\{\binom{n}{i}\}$ by $\{b_n^i\}_{n\geq 0}$. Then, we have that

$$u_j = \sum_{i=0}^{LC-1} c_i b_j^i, \quad j \geq 0$$

$$u_j + u_{j+1} = \sum_{i=0}^{LC-1} c_i b_j^i + \sum_{i=0}^{LC-1} c_i b_{j+1}^i = \sum_{i=0}^{LC-1} c_i \left(b_j^i + b_{j+1}^i\right), \tag{29}$$

$$j \geq 0.$$

According to Proposition 11,

$$u_j + u_{j+1} = \sum_{i=1}^{LC-1} c_i b_j^{i-1} = \sum_{k=0}^{LC-2} c_{k+1} b_j^k \tag{30}$$

Then, $\{u_j + u_{j+1}\}_{j\geq 0}$ is represented by $\sum_{i=0}^{LC-2} c_{i+1} \{\binom{n}{i}\}$. □

Remark 19. If the term $\binom{n}{0}$ is included in the binomial representation, it is discarded for the next sequence. See, for example, Table 4. In this table, we have two examples of one-dimensional linear CAs. The first one is a 102-CA. At the bottom of the CA, we can observe the binomial representations of the generated vertical sequences. We can check that the binomial representations of the sequences can be obtained following the process mentioned in Theorem 18. It is worth noticing that the given 60-CA generates exactly the same sequences, but they appear in reverse order.

Finally, observe that the set of binomial representations of a sequence follows the same pattern as the 102-CA.

Theorem 20 ([13], Theorem 4). *Given a sequence with period* 2^L *and linear complexity LC, then the CA that generates this sequence using the rule 102 has*

(i) *one sequence of period 1 (the identically 1 sequence),*

(ii) 2^{i-1} *sequences of period* 2^i*, for* $1 \leq i \leq L-2$,

(iii) $\mathrm{LC} - 2^{L-2}$ *sequences of period* 2^{L-1}.

Consider, for example, the sequence represented by $\{\binom{n}{0} + \binom{n}{2} + \binom{n}{3} + \binom{n}{5} + \binom{n}{12}\}$. This sequence has period $T = 16$. In Table 5, we can observe the 16 different representations of this sequence. The rows of the table represent the coefficients $\{c_0, c_1, \ldots, c_{12}\}$ that accompany each binomial coefficient, for each representation. That means, the column j represents the coefficients that accompany $\{\binom{n}{j}\}$ for each one of the 16 representations.

If we observe the behavior of the coefficients in the columns, we can check that the columns follow the same structure proposed in Theorem 20:

(i) One sequence of period 1 (rightmost sequence).

(ii) One sequences of period 2.

(iii) Two sequences of period 4.

(iv) Four sequences of period 8.

(v) Five sequences of period 16.

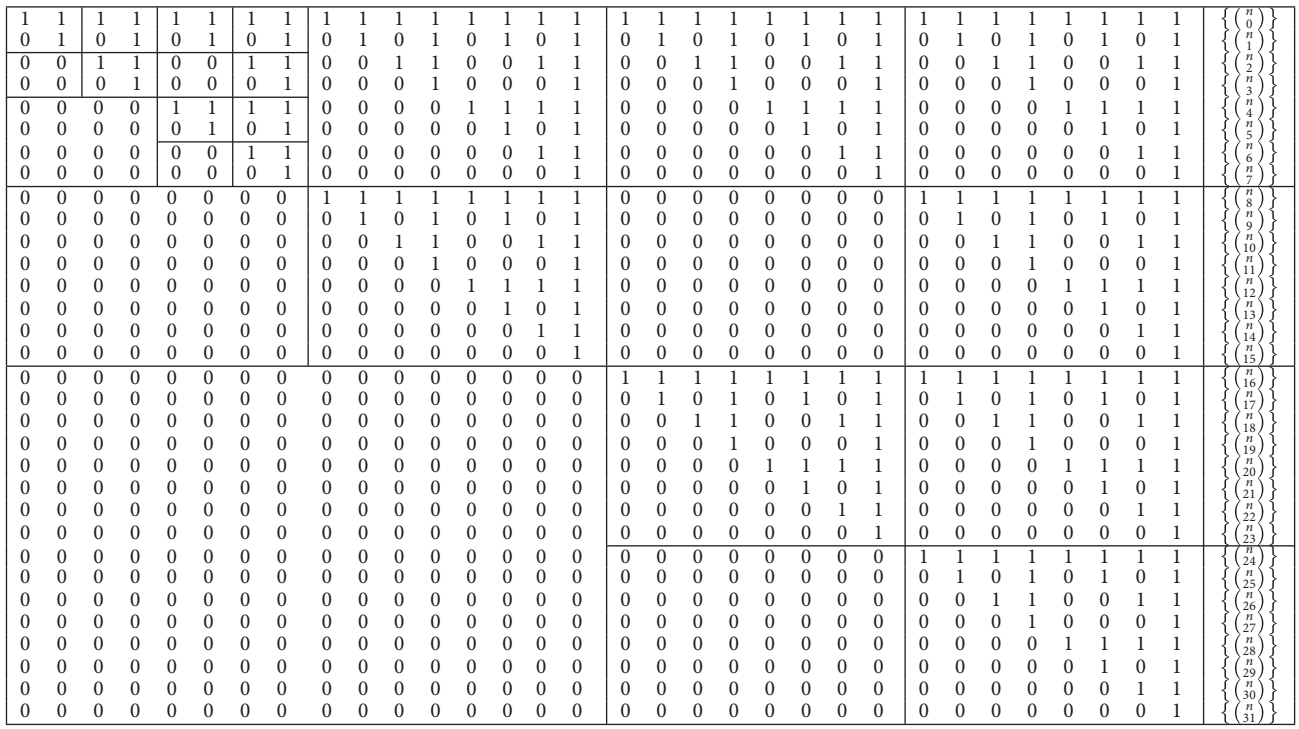


Figure 8: The first 32 binomial sequences.

Table 4: Examples of 102-CA and 60-CA and the binomial representation of the generated vertical sequences.

(a)

102	102	102	102	102
0	0	1	0	1
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	1	1	1	1
0	0	0	0	1
0	0	0	1	1
$\left\{\binom{n}{4}+\binom{n}{2}\right\}$	$\left\{\binom{n}{3}+\binom{n}{1}\right\}$	$\left\{\binom{n}{2}+\binom{n}{0}\right\}$	$\left\{\binom{n}{1}\right\}$	$\left\{\binom{n}{0}\right\}$

(b)

60	60	60	60	60
1	0	1	0	0
1	1	1	1	0
1	0	0	0	1
1	1	0	0	1
1	0	1	0	1
1	1	1	1	1
1	0	0	0	0
1	1	0	0	0
$\left\{\binom{n}{0}\right\}$	$\left\{\binom{n}{1}\right\}$	$\left\{\binom{n}{2}+\binom{n}{0}\right\}$	$\left\{\binom{n}{3}+\binom{n}{1}\right\}$	$\left\{\binom{n}{4}+\binom{n}{2}\right\}$

Table 5: Binomial representations of the 16 shifted versions of the sequence $\{\binom{n}{0} + \binom{n}{2} + \binom{n}{3} + \binom{n}{5} + \binom{n}{12}\}$.

$\binom{n}{0}$	$\binom{n}{1}$	$\binom{n}{2}$	$\binom{n}{3}$	$\binom{n}{4}$	$\binom{n}{5}$	$\binom{n}{6}$	$\binom{n}{7}$	$\binom{n}{8}$	$\binom{n}{9}$	$\binom{n}{10}$	$\binom{n}{11}$	$\binom{n}{12}$
c_0	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9	c_{10}	c_{11}	c_{12}
1	0	1	1	0	1	0	0	0	0	0	0	1
1	1	0	1	1	1	0	0	0	0	0	1	1
0	1	1	0	0	1	0	0	0	0	1	0	1
1	0	1	0	1	1	0	0	0	1	1	1	1
1	1	1	1	0	1	0	0	1	0	0	0	1
0	0	0	1	1	1	0	1	1	0	0	1	1
0	0	1	0	0	1	1	0	1	0	1	0	1
0	1	1	0	1	0	1	1	1	1	1	1	1
1	0	1	1	1	1	0	0	0	0	0	0	1
1	1	0	0	0	1	0	0	0	0	0	1	1
0	1	0	0	1	1	0	0	0	0	1	0	1
1	1	0	1	0	1	0	0	0	1	1	1	1
0	1	1	1	1	1	0	0	1	0	0	0	1
1	0	0	0	0	1	0	1	1	0	0	1	1
1	0	0	0	1	1	1	0	1	0	1	0	1
1	0	0	1	0	0	1	1	1	1	1	1	1
$T = 16$	$T = 16$	$T = 16$	$T = 16$	$T = 16$	$T = 8$	$T = 8$	$T = 8$	$T = 8$	$T = 4$	$T = 4$	$T = 2$	$T = 1$

Furthermore, it is possible to check that Table 5 is a 102-CA. This is due to the formation rule of the binomial representations given in (26), which coincides with the formation procedure of Rule 102.

6. Recovering the Binomial Representation

Given t intercepted bits of a sequence of period 2^L, Algorithm 1 introduces a method to recover a part of the binomial representation of such a sequence depending on the number t. Let us denote by $\boldsymbol{s}$ the set of intercepted bits. In round j, the algorithm compares $\boldsymbol{s}_j$ with the corresponding bit in the sequence represented by $\sum_{i=0}^{j} \{\binom{n}{i}\}$. If they match, then $\{\binom{n}{j}\}$ is part of the binomial representation. Otherwise, the term $\{\binom{n}{j}\}$ is discarded and the algorithm continues. This method is based on the fact that the first j bits of the sequence represented by $\{\binom{n}{j}\}$ are 0s.

Let us introduce now an illustrative example.

Example 21. Consider the set of intercepted bits $\boldsymbol{s} = \{1\ 1\ 0\ 1\ 1\ 1\}$. The first two bits ($\boldsymbol{s}_0 = 1$ and $\boldsymbol{s}_1 = 1$) match with the first two bits of the sequence $\{\binom{n}{0}\} = \{\mathbf{1}\ \mathbf{1}\ 1\ 1\ \ldots\}$. This means that one of the binomial representations of the sequence starts with $\sum_{i=0}^{1} c_i \{\binom{n}{i}\} = \{\binom{n}{0}\}$ $(c_0 = 1, c_1 = 0)$.

The bit $\boldsymbol{s}_2 = 0$ matches with the corresponding bit of the sequence $\{\binom{n}{0} + \binom{n}{2}\}$:

$$\begin{array}{l} \left\{\binom{n}{0}\right\} : 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ \ldots \\ \left\{\binom{n}{2}\right\} : 0\ 0\ 1\ 1\ 0\ 0\ 1\ 1\ \ldots \\ \hline 1\ 1\ \mathbf{0} \end{array} \tag{31}$$

Then, the binomial representation we are considering starts with $\sum_{i=0}^{2} c_i \{\binom{n}{i}\} = \{\binom{n}{0} + \binom{n}{2}\}$ $(c_0 = 1, c_1 = 0, c_2 = 1)$.

Finally, the bits $\boldsymbol{s}_3 = 1$, $\boldsymbol{s}_4 = 1$, and $\boldsymbol{s}_5 = 1$ match with the corresponding bits of the sequence $\{\binom{n}{0} + \binom{n}{2} + \binom{n}{3}\}$:

$$\begin{array}{l} \left\{\binom{n}{0}\right\} : 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ \ldots \\ \left\{\binom{n}{2}\right\} : 0\ 0\ 1\ 1\ 0\ 0\ 1\ 1\ \ldots \\ \left\{\binom{n}{3}\right\} : 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ \ldots \\ \hline 1\ 1\ 0\ \mathbf{1}\ \mathbf{1}\ \mathbf{1} \end{array} \tag{32}$$

Therefore, we have that the first part of the considered binomial representation is $\sum_{i=0}^{5} c_i \{\binom{n}{i}\} = \{\binom{n}{0} + \binom{n}{2} + \binom{n}{3}\}$ with coefficients as in Table 6. In case of having more intercepted bits and proceeding in the same way, we would complete the whole representation.

Next, we introduce a result on the number of bits required to recover the binomial representation of a sequence. Notice that if we know the binomial representation of a sequence, we can recover the whole sequence.

Proposition 22. *Given $LC - 1$ intercepted bits of a sequence with linear complexity LC and period 2^L, it is possible to recover the complete the binomial representation of the sequence.*

Proof. According to Corollary 14, the binomial representation of a sequence with linear complexity LC and period 2^L is of the form $\sum_{i=0}^{LC-1} c_i \{\binom{n}{i}\}$, with $c_{LC-1} = 1$. Now, according to the method explained in Algorithm 1, we need $LC - 1$ bits to recover each one of the coefficients c_i, $i = 0, 1, \ldots, LC-1$. □

Table 6: Coefficients in Example 21.

c_0	c_1	c_2	c_3	c_4	c_5
1	0	1	1	0	0

Input:
$\boldsymbol{s}$: Intercepted bits
01: $\boldsymbol{repr}$ = {0 0 0 0 ... };
02: $t = length(\boldsymbol{v})$;
03: **for** j = 0 **to** $t - 1$ **do**
04: **if** $\boldsymbol{s}_j \neq \boldsymbol{repr}_j$ then
05: $\boldsymbol{repr} = \boldsymbol{repr} + \binom{n}{j}$;
06: **endif**
07: **endfor**
Output:
$\boldsymbol{repr}$: Binomial representation of the intercepted bits

Algorithm 1: Constructing the binomial representation of a given sequence.

At any rate, the application of the traditional Berlekamp-Massey algorithm [16] needs $2 \cdot LC$ intercepted bits to recover the whole sequence. Thus, the method here developed makes use of half the bits needed by the Berlekamp-Massey algorithm. Consequently, the amount of intercepted bits has been reduced by a factor 2, which is quite favorable in terms of cryptanalysis.

7. Conclusions

The family of binary sequences considered in this work, sequences whose period is a power of 2, has good cryptographic properties such as long period and large linear complexity. However, we have seen that such sequences are simple solutions of linear difference equations with constant coefficients and can be obtained by XORing binomial binary sequences corresponding to diagonals of Sierpinski's triangle reduced modulo 2. Although different nonlinear procedures, e.g., irregular decimation, are introduced to break the linearity of the LFSR-based sequence generators, this linearity is still visible in their output sequences. Consequently, such linearity makes the generators producing the previous sequences vulnerable against cryptanalysis and makes them not suitable as part of more complex cryptographic structures. In this sense, we conjecture that given a sequence there exists a minimal binomial representation, that is, a representation with a minimum number of binomial terms.

On the other hand, we showed that there exists a close relation between one-dimensional linear cellular automata (102-CAs or 60-CAs) and the binomial sequences. Furthermore, there exists another family of cellular automata (150/90-CAs) that also generate sequences of period 2^L with good cryptographic properties. Therefore, in order to complete this study, the analysis of the relation of this family of cellular automata with binomial sequences is proposed as future work.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research has been partially supported by Ministerio de Economía, Industria y Competitividad (MINECO), Agencia Estatal de Investigación (AEI), and Fondo Europeo de Desarrollo Regional (FEDER, UE) under Project COPCIS, Reference TIN2017-84844-C2-1-R, and by Comunidad de Madrid (Spain) under Project Reference CYNAMON (P2018/TCS-4566) and also cofunded by European Union FEDER funds. The first author was supported by CAPES (Brazil). Finally, we would also like to thank Dr. Verónica Requena for her useful comments and suggestions.

References

[1] S. W. Golomb, *Shift Register-Sequences*, Aegean Park Press, Laguna Hill, California, USA, 1982.

[2] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, Boca Raton, FL, USA, 1996.

[3] W. Meier and O. Staffelbach, "The self-shrinking generator," in *Advances in cryptology -EUROCRYPT '94 (Perugia)*, A. De Santis, Ed., vol. 950 of *Lecture Notes in Computer Science*, pp. 205–214, Springer, Berlin, Heidelberg, Germany, 1995.

[4] Y. Hu and G. Xiao, "Generalized self-shrinking generator," *Institute of Electrical and Electronics Engineers Transactions on Information Theory*, vol. 50, no. 4, pp. 714–719, 2004.

[5] A. Kanso, "Modified self-shrinking generator," *Computers and Electrical Engineering*, vol. 36, no. 5, pp. 993–1001, 2010.

[6] S. D. Cardell and A. Fúster-Sabater, "The t-modified self-shrinking generator," in *Computational Science – ICCS 2018*, Y. Shi, H. Fu, and Y. Tian, Eds., vol. 10860 of *Lecture Notes in Computer Science*, pp. 653–663, Springer International Publishing, Cham, 2018.

[7] A. Fauré, A. Naldi, C. Chaouiya, and D. Thieffry, "Dynamical analysis of a generic Boolean model for the control of the mammalian cell cycle," *Bioinformatics*, vol. 22, no. 14, pp. e124–e131, 2006.

[8] D. Zheng, G. Yang, X. Li, Z. Wang, F. Liu, and L. He, "An efficient algorithm for computing attractors of synchronous and asynchronous boolean networks," *Plos One*, vol. 8, no. 4, Article ID e60593, 2013.

[9] J. A. Aledo, S. Martínez, and J. C. Valverde, "Graph dynamical systems with general boolean states," *Applied Mathematics Information Sciences*, vol. 9, no. 4, pp. 1803–1808, 2015.

[10] J. A. Aledo, S. Martínez, F. L. Pelayo, and J. C. Valverde, "Parallel discrete dynamical systems on maxterm and minterm boolean functions," *Mathematical and Computer Modelling*, vol. 55, no. 3-4, pp. 666–671, 2012.

[11] J. A. Aledo, S. Martínez, and J. C. Valverde, "Parallel dynamical systems over graphs and related topics: a survey," *Journal of Applied Mathematics*, vol. 2015, Article ID 594294, 14 pages, 2015.

[12] A. Fúster-Sabater and P. Caballero-Gil, "Linear cellular automata as discrete models for generating cryptographic sequences," *Journal of Research and Practice in Information Technology*, vol. 40, no. 4, pp. 47–52, 2008.

[13] S. D. Cardell and A. Fúster-Sabater, "Linear models for the self-shrinking generator based on CA," *Journal of Cellular Automata*, vol. 11, no. 2-3, pp. 195–211, 2016.

[14] A. K. Das, A. Ganguly, A. Dasgupta, S. Bhawmik, and P. P. Chaudhuri, "Efficient characterisation of cellular automata," *IEE Proceedings Part E Computers and Digital Techniques*, vol. 137, no. 1, pp. 81–87, 1990.

[15] S. Wolfram, "Cellular automata as simple self-organizing system," *Caltrech preprint CALT*, pp. 68–938, 1982.

[16] J. L. Massey, "Shift-register synthesis and BCH decoding," *IEEE Transactions on Information Theory*, vol. 15, no. 1, pp. 122–127, 1969.

Hindawi
Complexity
Volume 2019, Article ID 6280960, 10 pages
https://doi.org/10.1155/2019/6280960

Research Article

Predecessors Existence Problems and Gardens of Eden in Sequential Dynamical Systems

Juan A. Aledo[1,2] **Luis G. Diaz**[1,2] **Silvia Martinez**[1,2] **and Jose C. Valverde**[1,2]

[1] *Institute of Applied Mathematics in Science and Engineering, Ciudad Real, Spain*
[2] *Department of Mathematics, University of Castilla-La Mancha, Albacete, Spain*

Correspondence should be addressed to Jose C. Valverde; jose.valverde@uclm.es

Received 4 December 2018; Accepted 18 February 2019; Published 7 March 2019

Academic Editor: Eric Campos-Canton

In this paper, we deal with one of the main computational questions in network models: the predecessor-existence problems. In particular, we solve algebraically such problems in sequential dynamical systems on maxterm and minterm Boolean functions. We also provide a description of the Garden-of-Eden configurations of any system, giving the best upper bound for the number of Garden-of-Eden points.

1. Introduction

Network models are the natural way to formalize the evolution of any phenomenon that includes cells, gens, entities, or any kind of distinct elements which interact among them. Thus, it is not amazing the big amount of papers, published in the last decades, associated with sciences (see, for instance, [1–18]) and engineering (see, for instance, [19–24]), where such models appear to represent different phenomena and study their behavior.

Usually, the possible states of the elements of a network model are represented by a (finite) number set K, which we will call *state set*. One of the most common cases occurs when such states are on-off, as happens in computer processes, where the number set representing such states values is the basic Boolean algebra $\{0, 1\}$. However, K could have a more complex structure as, for example, a general Boolean algebra with 2^n elements, for $n \in \mathbb{N}$ (see [25]).

The interaction among the elements of the phenomena is mathematically modeled by a graph called *dependency* or *wiring* graph. In such a graph, the elements are formalized by nodes and an edge between two nodes represents some kind of relation. When the relations are only unidirectional, they are usually named *influences* and the graph is directed (see, for instance, [26]), with arcs instead of edges. Commonly, the graphs employed in such models are *combinatorial*; that is, they have no multiple edges or arcs and they are also loop-free.

The evolution of the phenomenon is normally modeled by an operator which consists of local functions that provide the evolution of any vertex, depending on the relation of such a vertex with other vertices. In view of this, these local functions are also called *vertex* functions [27]. When the vertex functions correspond to the restriction of the operator to every vertex and their related ones, the model is said to be *homogeneous*. Other less restricted possibility is that the vertex functions are independent as, for instance, in [28]. When the state set is a Boolean algebra, the vertex functions are given by Boolean functions.

The interactions among elements of a phenomenon do typically not occur simultaneously. When it occurs, it is said that the model *updates synchronously* or *parallelly* (see [25, 26, 28–36]); otherwise it is said that the model updates *asynchronously* or *sequentially* (see [20, 27, 37–41]). In this last case, an *update order* is needed to specify the sequence in which the states of the elements evolve. The update order is usually formalized by permutations or finite words over the vertex set of the corresponding wiring graph.

The above four features (i.e., state set, wiring graph, local functions, and update order) determine the most of network models. But, to get a fair model, one should also think if the phenomenon under modelization has some random aspects,

as happens, for example, in an infection model, that is, if the system is *stochastic* or, in contrast, it is *deterministic*. Stochastic network models can be obtained by choosing relations among elements, local functions, or update orders randomly at every (iteration) time t.

In this work, we deal with deterministic network models which are discrete-time dynamical systems. In particular, we focus on models for which the state set is $\{0, 1\}$; the wiring graph could be any combinatorial one; the local functions are restriction of a global operator, given by a maxterm or minterm Boolean function; and the update order is a vertex-indexed permutation. These models will be referred to as *sequential dynamical systems* (SDS) on maxterm and minterm Boolean functions.

SDS are useful to model phenomena (as evolution of electrical power networks, disease propagation across a social contact network, traffic in large urban areas, transport computations, etc.) where causal dependencies among the elements exist (see [27]). In fact, *parallel dynamical systems* (PDS) can be seen as a particular case of such models by means of the principle of doubling the graph as shown in [27]. Moreover, SDS generalize the concept of the pioneers network models given by the original cellular automata (CA) [42] or Boolean networks (BN) [14]. (The abbreviations SDS, PDS, CA, BN, and GOE will be written for the singular and plural forms of the corresponding terms, since it seems better from an aesthetic point of view).

Once a SDS is set out, to know the evolution of a phenomenon under modelization, we need to describe the phase portrait of such a dynamical system in relation to the state set, the wiring graph, the local functions, and the update order. The phase portrait is always a directed graph, consisting of disconnected cycles that could be reached by trees. The system states in such cycles are called *periodic*, while the rest of them are called *eventually periodic*. In particular, the leaves of the trees are said to be *Gardens of Eden* (GOE), while the rest of the system states in the trees are called *transient* states. In view of this, to know the dynamics, we need to derive as much information as possible about the periodic structure of the SDS and their transient and GOE points.

The periodic structure of SDS on maxterm and minterm Boolean functions is studied in [37], where it is proved that any period can appear in their phase portrait, although fixed points cannot coexist with other periods. This issue is someway a generalization of the results in PDS on the same Boolean functions [29, 33], where only fixed points and 2-periodic orbits can appear but not coexist. This generalization on the periodic structure, obtained when passing from PDS to SDS, motivates the study of the behavior of the rest of the states for SDS that we perform in this work.

Given a SDS, the dynamics of the rest of the states can be obtained computationally by brute-force, using algorithms like those in [34, 40]. But, unfortunately, the results in such computations are only valid for such a SDS. Thus, in this work, we try to extract information (algebraically) about the dynamics of such states by exploring the properties of the SDS in relation to its four fundamental features.

The *predecessor-existence problems*, whose solution gives the GOE and transient points, have been traditionally considered as a computational question together with other ones as the *reachability problem* or the *permutation-existence problem* (see [27, 38, 39, 41, 43, 44]). Jointly with the predecessor existence problem (PRE), other related problems are often studied as the unique predecessor problem (UPRE), the coexistence of predecessors problem (APRE), and the number of predecessors problem (#PRE) [43].

As for the periodic structure, the results here obtained for SDS in relation to the existence of predecessors and GOE not only suppose an algebraic solution for such problems, but they constitute a generalization to those achieved for PDS in [31].

This paper is organized as follows. Section 2 presents some basic notations and background on SDS. In Section 3, the PRE problem is solved for SDS on maxterm and minterm Boolean functions. In particular, we provide a *fundamental predecessor* which exists when the state has predecessors. As a consequence, we give conditions that characterize GOE and provide bounds for the number of such states. Mathematical conditions for the UPRE and APRE problems are also given. Besides, we provide upper bounds for the #PRE problem which are the best possible. Finally, Section 4 shows some conclusions and future research directions.

2. Preliminaries

In this section, some fundamental concepts about sequential dynamical systems (SDS) are revised. For further reading, more details on this topic can be found in [27] and the references therein.

A sequential dynamical system is given by three basic elements associated with the state set of the entities:

(i) A dependency graph modelling the relations among the entities.

(ii) A global function showing the influence of the states of the entities in the evolution.

(iii) A permutation of the vertices of the dependency graph that indicates the order in which the entities are updated.

Let us proceed with a contextualization of such elements into this type of system.

The dependency graph is an undirected graph $G = (V, E)$, where $V = \{1, \ldots, n\}$ is the vertex set and each element i belonging to V is called a vertex or entity of the system, and E is the edge set containing the adjacency relationships among the entities. Each one of these vertices can be activated or deactivated, and it will be represented for the entity $i \in V$, respectively, as $x_i = 1$ or $x_i = 0$, being these values the associated state values of the entities.

It will be assumed along this document that G is connected because, otherwise, the results can be generalized simply by working on each connected component.

For each entity $i \in V$, as in [31], let us consider the set $A_G(i) \subseteq V$ as

$$A_G(i) = \{j \in V : \{i, j\} \in E\} \cup \{i\} \tag{1}$$

that is, the set of all the entities that interfere with i in its update. Similarly, for $W \subseteq V$, we denote

$$A_G(W) = \bigcup_{i \in W} A_G(i) \tag{2}$$

Also, let us consider the following sets:

$$\begin{aligned} A_G^*(i) &= A_G(i) \setminus \{i\} \\ A_G^*(W) &= A_G(W) \setminus W \end{aligned} \tag{3}$$

We will denote the complementary of a set by using the upper script ^{c}. For example, $A_G^c(W)$ will be the subset of entities belonging to V which do not belong to $A_G(W)$.

The update (or evolution) of the system is given by local functions $\{f_i\}_{i \in V}$, for each $i \in V$. Although these functions may be independent [28], in this document we will assume that they are the restriction of a global function f over $A_G(i)$. This kind of systems is called homogeneous.

In particular, this global evolution operator can be described from a Boolean function of n variables:

$$F : \{0,1\}^n \longrightarrow \{0,1\}^n \tag{4}$$

where the evaluation $F(x_1, \ldots, x_n)$ is computed from the values $x_1, \ldots, x_n \in \{0,1\}$ using the logical operators AND, OR, and NOT.

Some relevant cases of Boolean functions are the maxterms and minterms. Recall that a maxterm (resp., minterm) of n variables is a Boolean function f such that

$$\begin{aligned} &f(x_1, \ldots, x_n) \\ &\quad = z_1 \vee \cdots \\ &\qquad \vee z_n \left(\text{resp. } f(x_1, \ldots, x_n) = z_1 \wedge \cdots \wedge z_n\right), \end{aligned} \tag{5}$$

being $z_i = x_i$ or $z_i = x_i'$.

Since the function f originates F, sometimes in this document the name of f can be assimilated by F.

Finally, the order relationship between the vertices of the dependency graph indicates the order in which the states of the entities are updated. It is given by a permutation on V, $\pi = \pi_1 \mid \pi_2 \mid \cdots \mid \pi_n$, where π_1 is the first entity whose state updates, π_2 the second one, and so on.

Summarizing, we have the following.

Definition 1. Let $G = (V, E)$ be an undirected graph with $V = \{1, \ldots, n\}$, $\pi = \pi_1 \mid \pi_2 \mid \cdots \mid \pi_n$ a permutation on V and

$$[F, \pi] = F_{\pi_n} \circ \ldots \circ F_{\pi_1} : \{0,1\}^n \longrightarrow \{0,1\}^n, \tag{6}$$

a Boolean function such that $F_{\pi_i} : \{0,1\}^n \longrightarrow \{0,1\}^n$ updates the state of the entity $\pi_i \in V$ considering the state values of the entities belonging to $A_G(i)$, keeping the other states unaltered. We will denote by $[G, F, \pi]$-SDS the sequential (discrete) dynamical system over G with evolution operator $[F, \pi]$.

Along this document, we will use indistinctly $[F, \pi]$ and F to mention the evolution operator since π is implicit in the evaluation of the Boolean function in this context of SDS.

The fundamental idea which motivates this document is that the key to understand a dynamical system is knowing its orbits and its phase diagram, i.e., the split of the state space into its orbits.

Definition 2. Let $[G, F, \pi]$-SDS be a sequential dynamical system and $x = (x_1, \ldots, x_n) \in \{0,1\}^n$. The orbit in the SDS starting at x is the subset of the state space $\{0,1\}^n$ given by

$$\begin{aligned} &Orb(x) \\ &= \left\{y = (y_1, \ldots, y_n) \in \{0,1\}^n : F^{(t)}(x) = y, \ \forall t \in \mathbb{N}\right\} \end{aligned} \tag{7}$$

The orbits in a SDS are sorted lists of states. When $F(x) = y$, y is called the successor of x. A configuration y can be successor of more than one state, which are its predecessors. When it happens, the system is called dissipative.

If there is no x such that $F(x) = y$, i.e., if y is a configuration without a predecessor, then y is called Garden of Eden (GOE) of the system.

Given a generic configuration y, we define the following subsets of V, which will be useful throughout this document.

Let us consider the split of V into the following sets:

$$\begin{aligned} V_0 &= \{i \in V : y_i = 0\} \\ V_1 &= \{i \in V : y_i = 1\} \end{aligned} \tag{8}$$

namely, V_0 (resp., V_1) is the set of deactivated (resp., activated) entities of y.

Additionally, let us now consider the following sets contained in $A_G(V_0)$:

$$\begin{aligned} P_0 &= \{i \in V : \exists j \in V_0 \text{ such that } \{i, j\} \in E, i = \pi_r, j \\ &\quad = \pi_s \text{ and } s < r\} \\ Q_0 &= \{i \in V : \exists j \in V_0 \text{ such that } \{i, j\} \in E, i = \pi_r, j \\ &\quad = \pi_s \text{ and } s > r\} \end{aligned} \tag{9}$$

In other words, each element i belonging to P_0 (resp., Q_0) is adjacent to a vertex $j \in V_0$ which is updated, according the order expressed in π, before (resp., after) i.

Similarly for V_1, we consider the sets P_1 and Q_1 contained in $A_G(V_1)$:

$$\begin{aligned} P_1 &= \{i \in V : \exists j \in V_1 \text{ such that } \{i, j\} \in E, i = \pi_r, \mathrm{j} \\ &\quad = \pi_s \text{ and } s < r\} \\ Q_1 &= \{i \in V : \exists j \in V_1 \text{ such that } \{i, j\} \in E, i = \pi_r, j \\ &\quad = \pi_s \text{ and } s > r\} \end{aligned} \tag{10}$$

Lastly, let us consider $W \subseteq V$ (resp., $W' \subseteq V$) the set of entities such that the corresponding variables appear in the Boolean operator, maxterm or minterm, in direct (resp., complemented) form. These sets W and W' are such that $W' = W^c$.

3. Main Results

In [43], the study of predecessors is divided into four specific problems:

(i) Predecessor existence (PRE).

(ii) Predecessor uniqueness (UPRE).

(iii) Predecessors coexistence (APRE).

(iv) Number of predecessors of a given state (#PRE).

We will start solving the first of these problems. We will also provide a characterization of GOE points in the context of SDS on maxterm and minterm Boolean functions. Therefrom, some results will be also reached which will allow us to solve the rest of the problems in the previous list.

The next theorem provides us with a characterization of GOE points in terms of sufficient and necessary conditions, finding a particular predecessor of a specific global state of the variables in the context of a SDS, when it exists.

Theorem 3. *Let* $[G, \mathrm{MAX}, \pi]$-SDS *be a sequential dynamical system over a dependency graph* $G = (V, E)$ *associated with the maxterm* MAX. *Then, a configuration* y *has a predecessor if, and only if, the state* x *defined as follows is such that* $\mathrm{MAX}(x) = y$*:*

(i) *For every entity* $i \in V_0 \cup P_0$

(a) $x_i = 0$ *if* $i \in W$,
(b) $x_i = 1$ *if* $i \in W'$.

(ii) *For every entity* $i \in (V_0 \cup P_0)^c$

(a) $x_i = 1$ *if* $i \in W$,
(b) $x_i = 0$ *if* $i \in W'$.

Proof. It must only be shown that this condition is necessary for the existence of a predecessor. For this purpose, let us see that if there is a predecessor of y, $\widehat{x} = (\widehat{x}_1, \ldots, \widehat{x}_n)$, then x defined as in this theorem is also a predecessor of y.

Thus, if $i \in V_0 \cup P_0$, it must be

(i) $\widehat{x}_i = 0 = x_i$ if $i \in W$,

(ii) $\widehat{x}_i = 1 = x_i$ if $i \in W'$;

otherwise $y_i = 1$ if $i \in V_0$ or $y_j = 1$ for some $j \in A_G^*(i) \cap V_0$ (those ones that update before i) if $i \in P_0$.

Suppose, by reduction to the absurd, that x is not a predecessor of y. Let $i \in V$ be the first entity, according to the order established by π, such that x_i does not update to y_i. It must be $i \in V_0 \cup P_0$, because the entities in $(V_0 \cup P_0)^c \subseteq V_1$ update to the activated state because of their own state values in x.

If $i \in P_0 \setminus V_0 \subseteq V_1$, let us analyze the possible state of the entities belonging to $A_G(i)$:

(i) Since i is the first entity not updating to the state given by $y_i = 1$, then $\forall j \in A_G^*(i)$ with $i = \pi_r$, $j = \pi_s$, and $s < r$, the entity j has updated to the state given by y_j, the same as for $\widehat{x}$,

(ii) $x_i = \widehat{x}_i$,

(iii) $\forall j \in A_G^*(i)$ with $i = \pi_r$, $j = \pi_s$, and $s > r$,

(a) if $j \in P_0 \cup V_0$, then $x_j = \widehat{x}_j$,
(b) if $j \in (P_0 \cup V_0)^c$, then $x_j = 1$ if $j \in W$ or $x_j = 0$ if $j \in W'$.

Since $\widehat{x}_i$ updates to $y_i = 1$, x_i must also do it, but this is a contradiction and, consequently, $i \notin P_0 \setminus V_0$.

Therefore $i \in V_0$. In this situation,

(i) Since i is the first entity not updating to the state given by $y_i = 0$, then $\forall j \in A_G^*(i)$ with $i = \pi_r$, $j = \pi_s$, and $s < r$, the entity j has updated to the state given by y_j, the same as for $\widehat{x}$,

(ii) $x_i = \widehat{x}_i$,

(iii) $\forall j \in A_G^*(i)$ with $i = \pi_r$, $j = \pi_s$, and $s > r$, the entity $j \in P_0$, and so $x_j = 0$ if $j \in W$ or $x_j = 1$ if $j \in W'$.

Since $\widehat{x}_i$ updates to $y_i = 0$, x_i must also do it, which is a contradiction and, consequently, $i \notin V_0$.

Therefore, there cannot exist $i \in V$ like that and x updates to y. □

Dually, we have the following.

Theorem 4. *Let* $[G, \mathrm{MIN}, \pi]$-SDS *be a sequential dynamical system over a dependency graph* $G = (V, E)$ *associated with the minterm* MIN. *Then, a configuration* y *has a predecessor if, and only if, the state* x *defined as follows is such that* $\mathrm{MIN}(x) = y$*:*

(i) *For every entity* $i \in V_1 \cup P_1$

(a) $x_i = 1$ *if* $i \in W$,
(b) $x_i = 0$ *if* $i \in W'$.

(ii) *For every entity* $i \in (V_1 \cup P_1)^c$

(a) $x_i = 0$ *if* $i \in W$,
(b) $x_i = 1$ *if* $i \in W'$.

Theorems 3 and 4 solve the PRE problem for SDS on maxterm and minterm Boolean functions as evolution operators, respectively, and allow us to establish the following characterization of the GOE points of these systems.

Corollary 5. *Let* $[G, \mathrm{MAX}, \pi]$-SDS *be a sequential dynamical system over a dependency graph* $G = (V, E)$ *associated with the maxterm* MAX. *Then, a configuration* y *is a GOE point of the system if, and only if, the state* x *defined as in Theorem 3 is such that* $\mathrm{MAX}(x) \neq y$.

Corollary 6. *Let* $[G, \mathrm{MIN}, \pi]$-SDS *be a sequential dynamical system over a dependency graph* $G = (V, E)$ *associated with the minterm* MIN. *Then, a configuration* y *is a GOE point of the system if, and only if, the state* x *defined as in Theorem 4 is such that* $\mathrm{MIN}(x) \neq y$.

Next, we provide sufficient conditions of GOE point.

Corollary 7. *Let* $[G, \mathrm{MAX}, \pi]$-SDS *be a sequential dynamical system over a dependency graph* $G = (V, E)$ *associated with the maxterm* MAX. *If a global state* y *is such that* $(Q_0 \cap V_0) \cap W' \neq \emptyset$, *then* y *is a GOE point.*

Proof. If $(Q_0 \cap V_0) \cap W' \neq \emptyset$, there is an entity $i \in V_0$ whose corresponding variable in MAX appears in complemented form and with an adjacent entity $j \in V_0$ updating after it. In this situation, the configuration y cannot be obtained as the update of another global state x because the evolution of the entity i to the deactivated state makes it impossible the posterior update of the entity j to this state. □

Corollary 8. *Let* $[G, \mathrm{MAX}, \pi]$-SDS *be a sequential dynamical system over a dependency graph* $G = (V, E)$ *associated with the maxterm* MAX. *If a global state* y *is such that* $(Q_0 \cap V_0^c) \cap W \neq \emptyset$, *then* y *is a* GOE *point.*

Proof. If $(Q_0 \cap V_0^c) \cap W \neq \emptyset$, there is an entity $i \in V_1$ whose corresponding variable in MAX appears in direct form and with an adjacent entity $j \in V_0$ updating after it. The proof finishes reasoning as in Corollary 7. □

And dually, we have the following.

Corollary 9. *Let* $[G, \mathrm{MIN}, \pi]$-SDS *be a sequential dynamical system over a dependency graph* $G = (V, E)$ *associated with the minterm* MIN. *If a global state* y *is such that* $(Q_1 \cap V_1) \cap W' \neq \emptyset$, *then* y *is a* GOE *point.*

Corollary 10. *Let* $[G, \mathrm{MIN}, \pi]$-SDS *be a sequential dynamical system over a dependency graph* $G = (V, E)$ *associated with the minterm* MIN. *If a global state* y *is such that* $(Q_1 \cap V_1^c) \cap W \neq \emptyset$, *then* y *is a* GOE *point.*

From Corollary 7 (resp., Corollary 9), it can be deduced that a configuration with two adjacent complemented vertices with state value 0 (resp., 1) is a GOE point of a $[G, \mathrm{MAX}, \pi]$ − SDS (resp., $[G, \mathrm{MIN}, \pi]$ − SDS). Actually, when MAX = NAND (resp., MIN = NOR), they are the only GOE points of the system.

Indeed, if a configuration y of $[G, \mathrm{NAND}, \pi]$ − SDS is such that $y_i = y_j = 0$ implies that $\{i, j\} \notin E$, then y has a predecessor given by

(i) $x_i = 1$ if $i \in V_0 \cup P_0$,

(ii) $x_i = 0$ if $i \in (V_0 \cup P_0)^c$.

Additionally, since in such kind of systems any configuration reaches a periodic orbit at a maximum of one iteration (see [36]), we have the following.

Proposition 11. *Let* $[G, \mathrm{NAND}, \pi]$ − SDS *be a sequential dynamical system over a dependency graph* $G = (V, E)$ *associated with the maxterm* NAND. *Then the* GOE *points of the system are the configurations such that there are two adjacent vertices* $i, j \in V$ *with* $x_i = x_j = 0$. *Furthermore, the other configurations belong to periodic orbits.*

And dually, we have the following.

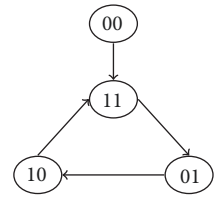


Figure 1: Phase diagram of the system.

Proposition 12. *Let* $[G, \mathrm{NOR}, \pi]$ − SDS *be a sequential dynamical system over a dependency graph* $G = (V, E)$ *associated with the minterm* NOR. *Then the* GOE *points of the system are the configurations such that there are two adjacent vertices* $i, j \in V$ *with* $x_i = x_j = 1$. *Furthermore, the other configurations belong to periodic orbits.*

From Proposition 11 (and, dually, from Proposition 12), we can see that the pattern of the parallel update shown in [31] for a MAX − PDS over a dependency graph $G = (V, E)$, with $V = \{1, \ldots, n\}$ and $n \geq 2$, whereby a configuration with only one activated entity has no predecessors, is broken in the case of the sequential update, as shown in the following example.

Example 13. In the case of the SDS defined by

(i) $G = \{\{1, 2\}, \{\{1, 2\}\}\}$,

(ii) $\mathrm{MAX} = x_1' \vee x_2'$,

(iii) $\pi = 1 \mid 2$,

the configuration $y = (0, 1)$ is not a GOE point because $x = (1, 1)$ is its predecessor.

In view of these results, we can state the following corollaries about the number of GOE points in a SDS.

Corollary 14. *Let* $[G, \mathrm{MAX}, \pi]$-SDS *be a sequential dynamical system over a dependency graph* $G = (V, E)$ *associated with the maxterm* MAX, *with* $V = \{1, \ldots, n\}$ *and* $n \geq 2$. *Then, the number of* GOE *points,* #GOE, *is such that* $1 \leq \#\mathrm{GOE} \leq 2^n - 2$. *Moreover, these bounds are the best possible because they are reachable.*

Proof. First, we will prove that any SDS with $n \geq 2$ has GOE points. Observe that, as $n \geq 2$, there exist two adjacent entities i and j with i updating before j. If $i \in W$, then a configuration with $y_i = 1$ and $y_j = 0$ has no predecessor; otherwise $i \in W'$ and the same occurs for a configuration with $y_i = y_j = 0$.

In fact, the lower bound is reached, as shown in the example below. Let us consider the $[G, \mathrm{MAX}, \pi]$-SDS defined by

(i) $G = (\{1, 2\}, \{\{1, 2\}\})$.

(ii) $\mathrm{MAX} = x_1' \vee x_2'$.

(iii) $\pi = 1 \mid 2$.

In this case, $(0, 0)$ is a GOE of the system and the rest of states belong to a 3-cycle, as can be checked in Figure 1.

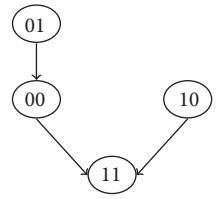


Figure 2: Phase diagram of the system.

On the other hand, $y = (1, \ldots, 1)$ is never a GOE of the system, because the state x defined as follows is its predecessor:

(i) $x_i = 1$ if $i \in W$,

(ii) $x_i = 0$ if $i \in W'$.

Also, there is always another point with a predecessor, because if $\overline{x}$ is defined as

(i) $\overline{x}_i = 0$ if $i \in W$,

(ii) $\overline{x}_i = 1$ if $i \in W'$

then $\overline{x}$ updates to a state $\overline{y}$ such that $\overline{y}_1 = 0$.

As shown in the example below, this upper bound is also reached. Let us consider the following $[G, \mathrm{MAX}, \pi]$-SDS, determined by

(i) $G = (\{1, 2\}, \{\{1, 2\}\})$.

(ii) $\mathrm{MAX} = x_1 \vee x_2'$.

(iii) $\pi = 1 \mid 2$.

The phase diagram of this system is shown in Figure 2. □

Remark 15. In Corollary 14, $n \geq 2$ has been imposed. This is necessary because a $[G, \mathrm{MAX}, \pi]$- SDS with $n = 1$ has 2 fixed points, if $W' = \emptyset$, or one 2-cycle, if $W = \emptyset$. That is, it has not GOE points in any case.

And dually, we have the following.

Corollary 16. *Let* $[G, \mathrm{MIN}, \pi]$-SDS *be a sequential dynamical system over a dependency graph* $G = (V, E)$ *associated with the minterm* MIN*, with* $V = \{1, \ldots, n\}$ *and* $n \geq 2$*. Then, the number of* GOE *points,* #GOE*, is such that* $1 \leq \#\mathrm{GOE} \leq 2^n - 2$*. Moreover, these bounds are the best possible because they are reachable.*

In Theorem 3, a constructive proof about the existence of a predecessor is shown. The structure of such a predecessor exposed in that result inspires the following result.

Corollary 17. *Let* $[G, \mathrm{MAX}, \pi]$-SDS *be a sequential dynamical system over a dependency graph* $G = (V, E)$ *associated with the maxterm* MAX*. If a configuration* y *has a predecessor* x*, the structure of such a predecessor is as follows:*

(i) *If* $y_i = 0$*, for every entity* $j \in A_G(i)$*, with* $i = \pi_r$ *and* $j = \pi_s$*:*

(a) *If* $i = j$ *or* $r < s$*:*

(1) $x_j = 0$ *if* $j \in W$*, or*

(2) $x_j = 1$ *if* $j \in W'$.

(b) *If* $r > s$*:*

(1) $y_j = 0$ *if* $j \in W$*, or*

(2) $y_j = 1$ *if* $j \in W'$.

(ii) *If* $y_i = 1$*, there exists an entity* $j \in A_G(i)$ *such that if* $i = \pi_r$ *and* $j = \pi_s$*, at least one of the following conditions is accomplished:*

(a) $i = j$ *or* $r < s$*, and:*

(1) $x_j = 1$ *if* $j \in W$*, or*

(2) $x_j = 0$ *if* $j \in W'$.

(b) $r > s$*, and:*

(1) $y_j = 1$ *if* $j \in W$*, or*

(2) $y_j = 0$ *if* $j \in W'$.

Proof. On the one hand, if $y_i = 0$ and there is $j \in A_G(i)$ such that the conditions shown are not satisfied in this case, the entity i will update to the activated state due to this adjacent entity j, which is a contradiction. On the other hand, if $y_i = 1$ and $\forall j \in A_G(i)$ these conditions are not satisfied, the entity i will update to the deactivated state, which is also a contradiction. □

Dually, we have the following.

Corollary 18. *Let* $[G, \mathrm{MIN}, \pi]$-SDS *be a sequential dynamical system over a dependency graph* $G = (V, E)$ *associated with the minterm* MIN*. If a configuration* y *has a predecessor* x*, the structure of such a predecessor is as follows:*

(i) *If* $y_i = 1$*, for every entity* $j \in A_G(i)$*, with* $i = \pi_r$ *and* $j = \pi_s$*:*

(a) *If* $i = j$ *or* $r < s$*:*

(1) $x_j = 1$ *if* $j \in W$*, or*

(2) $x_j = 0$ *if* $j \in W'$.

(b) *If* $r > s$*:*

(1) $y_j = 1$ *if* $j \in W$*, or*

(2) $y_j = 0$ *if* $j \in W'$.

(ii) *If* $y_i = 0$*, there exists an entity* $j \in A_G(i)$ *such that if* $i = \pi_r$ *and* $j = \pi_s$*, at least one of the following conditions is accomplished:*

(a) $i = j$ *or* $r < s$*, and:*

(1) $x_j = 0$ *if* $j \in W$*, or*

(2) $x_j = 1$ *if* $j \in W'$.

(b) $r > s$*, and:*

(1) $y_j = 0$ *if* $j \in W$*, or*

(2) $y_j = 1$ *if* $j \in W'$.

In a MAX-SDS (resp., MIN-SDS), the entities whose state is deactivated (resp., activated) in y determine univocally their state and the state of their adjacent entities in P_0 (resp., P_1) in any predecessor x, if such a predecessor exists.

However, for any entity whose state value is 1 (resp., 0) in y, it is only necessary the intervention of a timely adjacent entity, or itself, with the appropriate state in the moment of its update. This point is the key to solve the UPRE, APRE, and #PRE problems hereafter.

The following theorems allow us to determine if, given a state y with a predecessor x, there are other configurations different from x such that they are also predecessors of y. Thus, the UPRE and APRE problems in the context of a SDS on maxterm and minterm Boolean functions as evolution operators are solved.

Theorem 19 (solution to the APRE and UPRE problems for maxterm-SDS). *Let* $[G, \mathrm{MAX}, \pi]$-SDS *be a sequential dynamical system over a dependency graph* $G = (V, E)$ *associated with the maxterm* MAX. *Let* y *be a configuration of the system such that it has a predecessor. Then, there is a unique predecessor if, and only if, there is not a predecessor of* y *belonging to the following set:*

$$\mathfrak{P} = \left\{\widehat{x} \in \{0,1\}^n : \exists i \in \left(V_0 \cup P_0\right)^c \text{ such that } \widehat{x}_i \neq x_i \text{ and } \widehat{x}_j = x_j \ \forall j \in V \setminus \{i\}\right\} \tag{11}$$

being x *the predecessor of* y *described in Theorem 3.*

Proof. Since x defined as in Theorem 3 is such that $x \notin \mathfrak{P}$, it must only be shown that this condition is sufficient for the existence of a unique predecessor. For this purpose, let us see that if there is a predecessor of y different from x, $\overline{x} = (\overline{x}_1, \ldots, \overline{x}_n)$, then there exists a state $\widehat{x} = (\widehat{x}_1, \ldots, \widehat{x}_n) \in \mathfrak{P}$ such that $\widehat{x}$ is also a predecessor of y.

Given that $\overline{x} \neq x$ and $\overline{x}$ is a predecessor of y, by Corollary 17, there is an entity $i_0 \in (V_0 \cup P_0)^c$ such that $\overline{x}_{i_0} \neq x_{i_0}$. Let us take $\widehat{x}$ as the only element of $\mathfrak{P}$ such that $\widehat{x}_{i_0} \neq x_{i_0}$ (consequently, $\widehat{x}_{i_0} = \overline{x}_{i_0}$), and let us see that this state is a predecessor of y.

Suppose, by reduction to the absurd, that $\widehat{x}$ is not a predecessor of y. Let $i \in V$ be the first entity, according to the order established by π, such that $\widehat{x}_i$ does not update to y_i. It must be $i \in V_0 \cup P_0 \cup \{i_0\}$, because the entities in $(V_0 \cup P_0 \cup \{i_0\})^c \subseteq V_1$ update to the activated state because of their own state values in $\widehat{x}$.

If $i \in P_0 \setminus (V_0 \cup \{i_0\}) = P_0 \setminus V_0 \subseteq V_1$, let us analyze the possible state of the entities belonging to $A_G(i)$:

(i) Since i is the first entity not updating to the state given by $y_i = 1$, then $\forall j \in A_G^*(i)$ with $i = \pi_r$, $j = \pi_s$, and $s < r$, the entity j has updated to the state given by y_j, the same as for $\overline{x}$,

(ii) $\widehat{x}_i = \overline{x}_i$,

(iii) $\forall j \in A_G^*(i)$ with $i = \pi_r$, $j = \pi_s$ and $s > r$,

(a) if $j \in P_0 \cup V_0 \cup \{i_0\}$ then $\widehat{x}_j = \overline{x}_j$,

(b) if $j \in (P_0 \cup V_0 \cup \{i_0\})^c$ then $\widehat{x}_j = 1$ if $j \in W$ or $\widehat{x}_j = 0$ if $j \in W'$.

Since $\overline{x}_i$ updates to $y_i = 1$, $\widehat{x}_i$ must also do it, but this is a contradiction and, consequently, $i \notin P_0 \setminus V_0$.

If $i \in V_0 \setminus \{i_0\} = V_0$, we have the following:

(i) Since i is the first entity not updating to the state given by $y_i = 0$, then $\forall j \in A_G^*(i)$ with $i = \pi_r$, $j = \pi_s$ and $s < r$, the entity j has updated to the state given by y_j, the same as for $\overline{x}$,

(ii) $\widehat{x}_i = \overline{x}_i$,

(iii) $\forall j \in A_G^*(i)$ with $i = \pi_r$, $j = \pi_s$, and $s > r$, the entity $j \in P_0 = P_0 \setminus \{i_0\}$, and so $x_j = 0$ if $j \in W$ or $x_j = 1$ if $j \in W'$.

Since $\overline{x}_i$ updates to $y_i = 0$, $\widehat{x}_i$ must also do it, which is a contradiction and, consequently, $i \notin V_0$.

Therefore $i = i_0$. In this situation, we have the following:

(i) Since i is the first entity not updating to the state given by $y_i = 1$ ($i_0 \in (V_0 \cup P_0)^c \subseteq V_1$), then $\forall j \in A_G^*(i_0)$ with $i_0 = \pi_r$, $j = \pi_s$, and $s < r$, the entity j has updated to the state given by y_j, the same as for $\overline{x}$,

(ii) $\widehat{x}_{i_0} = \overline{x}_{i_0}$,

(iii) $\forall j \in A_G^*(i_0)$ with $i_0 = \pi_r$, $j = \pi_s$ and $s > r$,

(a) if $j \in P_0 \cup V_0$ then $\widehat{x}_j = \overline{x}_j$,

(b) if $j \in (P_0 \cup V_0)^c$ then $\widehat{x}_j = 1$ if $j \in W$ or $\widehat{x}_j = 0$ if $j \in W'$.

Since $\overline{x}_i$ updates to $y_i = 1$, $\widehat{x}_i$ must also do it, but this is also a contradiction and, consequently, $i \neq i_0$.

Therefore, there cannot exist $i \in V$ like that and $\widehat{x}$ updates to y. □

Remark 20. The previous result reduces an initial exponentially sized problem, the search of a particular configuration among the 2^n possible states of the system, into another one in which, at most, n cases must be analyzed. In this case, a short list of possible candidates is provided and the evaluation of the evolution operation only over the elements of this set provides the answer to the global problem of existence of a unique predecessor for a state y.

And dually, we have the following.

Theorem 21 (solution to the APRE and UPRE problems for minterm-SDS). *Let* $[G, \mathrm{MIN}, \pi]$-SDS *be a sequential dynamical system over a dependency graph* $G = (V, E)$ *associated with the minterm* MIN. *Let* y *be a configuration of the system such that it has a predecessor. Then, there is a unique predecessor if, and only if, there is not a predecessor of* y *belonging to the following set:*

$$\mathfrak{P} = \left\{\widehat{x} \in \{0,1\}^n : \exists i \in \left(V_1 \cup P_1\right)^c \text{ such that } \widehat{x}_i \neq x_i \text{ and } \widehat{x}_j = x_j \ \forall j \in V \setminus \{i\}\right\} \tag{12}$$

being x *the predecessor of* y *described in Theorem 4.*

These results respond to the question of the existence of more than one predecessor for a state y. The next step is to

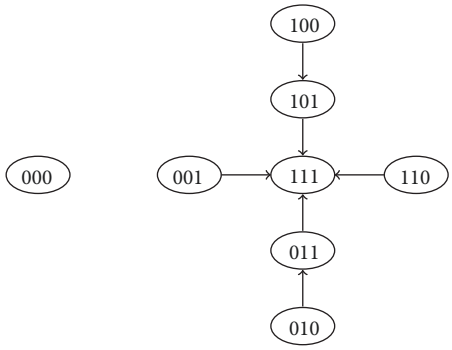


Figure 3: Phase diagram of the system.

go deeper into this topic, getting the number of them. In the following results, we explain a method to obtain all the predecessors of y and, consequently, this number in order to solve the predecessor problem #PRE.

Corollary 22. *Let* $[G, \mathrm{MAX}, \pi]$-SDS *be a sequential dynamical system over a dependency graph* $G = (V, E)$ *associated with the maxterm* MAX. *Let* y *be a configuration and let us consider the following iterative process:*

(i) $\mathfrak{P}_{n+1} = \{y\}$,

(ii) *if* $i \in V$, *then* $\mathfrak{P}_i = \{x \in \{0,1\}^n : \exists \overline{y} \in \mathfrak{P}_{i+1}$ *such that* $x_j = \overline{y}_j$ *if* $j \neq \pi_i$ *and* $\mathrm{MAX}_{|A_G(\pi_i)}(x) = y_{\pi_i}\}$, *being* $\mathrm{MAX}_{|A_G(\pi_i)}$ *the restriction of* MAX *over* $A_G(\pi_i)$.

Then, $\mathfrak{P}_1$ *is the set of all the predecessors of* y.

Example 23. Let us illustrate this procedure with a particular example, in order to clarify the notation. We consider the $[G, \mathrm{MAX}, \pi]$- SDS defined by

(i) $G = (\{1,2,3\}, \{\{1,3\}, \{2,3\}\})$.

(ii) $\mathrm{MAX} = x_1 \vee x_2 \vee x_3$.

(iii) $\pi = 1 \mid 2 \mid 3$.

The phase diagram of this system is shown in Figure 3.

The set of predecessors of the configuration $y = (1,1,1)$ is

$$\{(1,1,1), (1,1,0), (0,0,1), (0,1,1), (1,0,1)\} \quad (13)$$

Let us see that this set is obtained as $\mathfrak{P}_1$ at the end of the iterative process starting with $\mathfrak{P}_4 = \{(1,1,1)\}$.

Firstly, $\mathfrak{P}_3$ is obtained: since $\pi_3 = 3$, the only configurations that can belong to $\mathfrak{P}_3$ are $(1,1,0)$ and $(1,1,1)$. On the other side, since $\mathrm{MAX}_{|A_G(3)}(1,1,0) = \mathrm{MAX}_{|A_G(3)}(1,1,1) = 1 = y_3$, it is $\mathfrak{P}_3 = \{(1,1,0), (1,1,1)\}$.

Then, to obtain $\mathfrak{P}_2$, since $\pi_2 = 2$ and considering the elements in $\mathfrak{P}_3$, the only configurations that can belong to $\mathfrak{P}_2$ are $(1,0,0)$, $(1,1,0)$, $(1,0,1)$, and $(1,1,1)$. Now, $\mathrm{MAX}_{|A_G(2)}(1,0,0) = 0 \neq y_2$ and $\mathrm{MAX}_{|A_G(2)}(1,1,0) = \mathrm{MAX}_{|A_G(2)}(1,0,1) = \mathrm{MAX}_{|A_G(2)}(1,1,1) = 1 = y_2$, so $\mathfrak{P}_2 = \{(1,1,0), (1,0,1), (1,1,1)\}$.

Finally, since $\pi_1 = 1$ and knowing the set $\mathfrak{P}_2$, the only configurations that can belong to $\mathfrak{P}_1$ are $(0,1,0)$, $(1,1,0)$, $(0,0,1)$, $(1,0,1)$, $(0,1,1)$, and $(1,1,1)$. Now, $\mathrm{MAX}_{|A_G(1)}(0,1,0)$ is equal to $0 \neq y_1$, while $\mathrm{MAX}_{|A_G(1)}(1,1,0)$, $\mathrm{MAX}_{|A_G(1)}(0,0,1)$, $\mathrm{MAX}_{|A_G(1)}(1,0,1)$, $\mathrm{MAX}_{|A_G(1)}(0,1,1)$, and $\mathrm{MAX}_{|A_G(1)}(1,1,1)$ are all equal to $1 = y_1$.

Thus, $\mathfrak{P}_1 = \{(1,1,0), (0,0,1), (1,0,1), (0,1,1), (1,1,1)\}$.

Dually, we have the following.

Corollary 24. *Let* $[G, \mathrm{MIN}, \pi]$-SDS *be a sequential dynamical system over a dependency graph* $G = (V, E)$ *associated with the minterm* MIN. *Let* y *be a configuration and let us consider the following iterative process:*

(i) $\mathfrak{P}_{n+1} = \{y\}$,

(ii) *if* $i \in V$, *then* $\mathfrak{P}_i = \{x \in \{0,1\}^n : \exists \overline{y} \in \mathfrak{P}_{i+1}$ *such that* $x_j = \overline{y}_j$ *if* $j \neq \pi_i$ *and* $\mathrm{MIN}_{|A_G(\pi_i)}(x) = y_{\pi_i}\}$, *being* $\mathrm{MIN}_{|A_G(\pi_i)}$ *the restriction of* MIN *over* $A_G(\pi_i)$.

Then, $\mathfrak{P}_1$ *is the set of all the predecessors of* y.

These last procedures allow us to know all the predecessors of a state y in a SDS and, consequently, the number of them. Since this calculus depends on the particular graph, in the following results we give a bound for this number.

Theorem 25. *Let* $[G, \mathrm{MAX}, \pi]$-SDS *be a sequential dynamical system over a dependency graph* $G = (V, E)$ *associated with the maxterm* MAX. *Then, the number of predecessors of a given state* y *is upper bounded by* $2^{\#(V_0 \cup P_0)^c}$. *Moreover, this bound is the best possible because it is reachable.*

Proof. From Theorem 3 and Corollary 17, the states of the entities belonging to $V_0 \cup P_0$ in a possible predecessor of y are fixed. Since the state values of the rest of entities are either 0 or 1, a first upper bound for the number of predecessors is $2^{\#(V_0 \cup P_0)^c}$.

This upper bound is the best possible because it is reached in the following example. Let us consider the $[G, \mathrm{MAX}, \pi]$-SDS defined by

(i) $G = \{V, E\}$, with $V = \{1, \ldots, n\}$, $n \geq 2$, and $E = \{\{2, i\} : i \in V \setminus \{2\}\}$.

(ii) $\mathrm{MAX} = x_1' \vee x_2 \vee x_3 \vee \cdots \vee x_n$.

(iii) $\pi = 1 \mid \ldots \mid n$.

In this context, if $y = (0, 1, \ldots, 1)$, then $V_0 = \{1\}$, $P_0 = \{2\}$, $Q_0 = \emptyset$, and $V_1 = \{2, \ldots, n\}$.

In any predecessor, x, it must be $x_1 = 1$ and $x_2 = 0$, but, in this case, all the other choices for the states of the rest of entities generate predecessors of y. □

And dually, we have the following.

Theorem 26. *Let* $[G, \mathrm{MIN}, \pi]$-SDS *be a sequential dynamical system over a dependency graph* $G = (V, E)$ *associated with the minterm* MIN. *Then, the number of predecessors of a given state* y *is upper bounded by* $2^{\#(V_1 \cup P_1)^c}$. *Moreover, this bound is the best possible because it is reachable.*

4. Conclusions and Future Research Directions

The results here obtained for SDS in relation to the predecessors existence problems and GOE not only suppose an algebraic solution for such problems, but also constitute a generalization to those achieved for PDS recently.

On the other hand, these results could reveal interesting interpretations for those phenomena modeled by these SDS, what gives an idea of their relevance.

Taking into account that the interactions among elements of a phenomenon could be (only) unidirectional, these results encourage us to know what happens for SDS over directed graphs as a future research direction.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

Juan A. Aledo was supported by Junta de Comunidades de Castilla-La Mancha grant FEDER SBPLY/17/18050/000493. Luis G. Diaz, Silvia Martinez, and Jose C. Valverde were supported by FEDER OP2014-2020 of Castilla-La Mancha (Spain) under the Grant GI20173946.

References

[1] F. D. Abraham, "A beginners guide to the nature and potentialities of dynamical and network theory II: a very brief comparison of discrete networks to continuous dynamical systems," *Chaos and Complexity Letters*, vol. 9, no. 2, pp. 1–18, 2015.

[2] C. Bisi and G. Chiaselotti, "A class of lattices and boolean functions related to the Manickam-Miklös-Singhi conjecture," *Advances in Geometry*, vol. 13, no. 1, pp. 1–27, 2013.

[3] G. Chiaselotti, T. Gentile, and P. A. Oliverio, "Parallel and sequential dynamics of two discrete models of signed integer partitions," *Applied Mathematics and Computation*, vol. 232, pp. 1249–1261, 2014.

[4] G. Cattaneo, G. Chiaselotti, A. Dennunzio, E. Formenti, and L. Manzoni, "Non uniform cellular automata description of signed partition versions of ice and sand pile models," in *Cellular Automata. ACRI*, J. Was, G. C. Sirakoulis, and S. Bandini, Eds., vol. 8751 of *Lecture Notes in Computer Science*, pp. 115–124, Springer, Cham, Switzerland, 2014.

[5] G. Cattaneo, G. Chiaselotti, P. A. Oliverio, and F. Stumbo, "A new discrete dynamical system of signed integer partitions," *European Journal of Combinatorics*, vol. 55, pp. 119–143, 2016.

[6] G. Cattaneo, M. Comito, and D. Bianucci, "Sand piles: from physics to cellular automata models," *Theoretical Computer Science*, vol. 436, pp. 35–53, 2012.

[7] G. Chiaselotti, G. Marino, P. A. Oliverio, and D. Petrassi, "A discrete dynamical model of signed partitions," *Journal of Applied Mathematics*, vol. 2013, Article ID 973501, 10 pages, 2013.

[8] B. Chopard and M. Droz, *Cellular Automata for Modeling Physics*, Cambridge University Press, Cambridge, UK, 1998.

[9] A. Deutsch and S. Dormann, *Cellular Automaton Modelling of Biological Pattern Formation*, Birkhäuser, Boston, Mass, USA, 2004.

[10] U. Dieckman, R. Law, and J. A. J. Metz, *The Geometry of Ecological Interactions: Simplifying Spatial Complexity*, Cambridge University Press, Cambridge, 2000.

[11] W. Dzwinel, R. Wcisło, D. A. Yuen, and S. Miller, "PAM: particle automata in modeling of multiscale biological systems," *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, vol. 26, no. 3, article 20, 2016.

[12] J. Hofbauer and K. Sigmund, *Evolutionary Games and Population Dynamics*, Cambridge University Press, Cambridge, UK, 2003.

[13] P. Hogeweg, "Cellular automata as a paradigm for ecological modeling," *Applied Mathematics and Computation*, vol. 27, no. 1, pp. 81–100, 1988.

[14] S. A. Kauffman, "Metabolic stability and epigenesis in randomly constructed genetic nets," *Journal of Theoretical Biology*, vol. 22, no. 3, pp. 437–467, 1969.

[15] L. B. Kier and P. G. Seybold, "Cellular automata modeling of complex biochemical," in *Encyclopedia of Complexity and Systems Science*, R. A. Meyers, Ed., pp. 848–865, Springer, New York, NY, USA, 2009.

[16] L. B. Kier, P. G. Seybold, and C.-K. Cheng, *Modeling Chemical Systems Using Cellular Automata*, Springer, New York, NY, USA, 2005.

[17] D. Scalise and R. Schulman, "Emulating cellular automata in chemical reaction–diffusion networks," *Natural Computing*, vol. 15, no. 2, pp. 197–214, 2016.

[18] Z. Toroczkai and H. Guclu, "Proximity networks and epidemics," *Physica A: Statistical Mechanics and Its Applications*, vol. 378, no. 1, pp. 68–75, 2007.

[19] N. L. Ackerman and C. E. Freer, "Graph turing machines," in *Proceedings of the WoLLIC 2017*, vol. 10388 of *Lecture Notes in Comput. Sci.*, pp. 1–13, Springer, Berlin.

[20] C. Defant, "Binary codes and period-2 orbits of sequential dynamical systems," *Discrete Mathematics & Theoretical Computer Science*, vol. 19, no. 3, p. 10, 2017.

[21] A. Fúster-Sabater and P. Caballero-Gil, "On the use of cellular automata in symmetric cryptography," *Acta Applicandae Mathematicae*, vol. 93, no. 1–3, pp. 215–236, 2006.

[22] W. Gao, J. L. Guirao, and H. Wu, "Two tight independent set conditions for fractional (g,f,m)-deleted graphs systems," *Qualitative Theory of Dynamical Systems*, vol. 17, no. 1, pp. 231–243, 2018.

[23] A. Ilachinski, *Cellular Automata: A Discrete Universe*, World Scientific, Singapore, 2001.

[24] F. Jian and S. Dandan, "Complex network theory and its application research on P2P networks," *Applied Mathematics and Nonlinear Sciences*, vol. 1, no. 1, pp. 45–52, 2016.

[25] J. A. Aledo, S. Martinez, and J. C. Valverde, "Graph dynamical systems with general Boolean states," *Applied Mathematics & Information Sciences*, vol. 9, no. 4, pp. 1803–1808, 2015.

[26] J. A. Aledo, S. Martinez, and J. C. Valverde, "Parallel dynamical systems over directed dependency graphs," *Applied Mathematics and Computation*, vol. 219, no. 3, pp. 1114–1119, 2012.

[27] H. S. Mortveit and C. M. Reidys, *An Introduction to Sequential Dynamical Systems*, Springer, New York, NY, USA, 2007.

[28] J. A. Aledo, S. Martinez, and J. C. Valverde, "Parallel discrete dynamical systems on independent local functions," *Journal of Computational and Applied Mathematics*, vol. 237, no. 1, pp. 335–339, 2013.

[29] J. A. Aledo, L. G. Diaz, S. Martinez, and J. C. Valverde, "On the periods of parallel dynamical systems," *Complexity*, vol. 2017, Article ID 7209762, 6 pages, 2017.

[30] J. A. Aledo, L. G. Diaz, S. Martinez, and J. C. Valverde, "Maximum number of periodic orbits in parallel dynamical systems," *Information Sciences*, vol. 468, pp. 63–71, 2018.

[31] J. A. Aledo, L. G. Diaz, S. Martinez, and J. C. Valverde, "Predecessors and garden-of-eden configurations in parallel dynamical systems on maxterm and minterm Boolean functions," *Journal of Computational and Applied Mathematics*, vol. 348, pp. 26–33, 2019.

[32] J. A. Aledo, L. G. Diaz, S. Martinez, and J. C. Valverde, "Solution to the predecessors and gardens-of-eden problems for synchronous systems over directed graphs," *Applied Mathematics and Computation*, vol. 370, pp. 22–28, 2019.

[33] J. A. Aledo, S. Martinez, F. L. Pelayo, and J. C. Valverde, "Parallel dynamical systems on maxterm and minterm Boolean functions," *Mathematical and Computer Modelling*, vol. 35, pp. 666–671, 2012.

[34] J. A. Aledo, S. Martinez, and J. C. Valverde, "Updating method for the computation of orbits in parallel and sequential dynamical systems," *International Journal of Computer Mathematics*, vol. 90, no. 9, pp. 1796–1808, 2013.

[35] J. A. Aledo, S. Martinez, and J. C. Valverde, "Parallel dynamical systems over graphs and related topics: a survey," *Journal of Applied Mathematics*, vol. 2015, Article ID 594294, 14 pages, 2015.

[36] C. L. Barrett, W. Y. Chen, and M. J. Zheng, "Discrete dynamical systems on graphs and Boolean functions," *Mathematics and Computers in Simulation*, vol. 66, no. 6, pp. 487–497, 2004.

[37] J. A. Aledo, L. G. Diaz, S. Martinez, and J. C. Valverde, "On periods and equilibria of computational sequential systems," *Information Sciences*, vol. 409-410, pp. 27–34, 2017.

[38] C. L. Barrett, H. B. Hunt III, M. V. Marathe et al., "Gardens of Eden and fixed points in sequential dynamical systems," *Discrete Mathematics & Theoretical Computer Science Proceedings*, pp. 95–110, 2001.

[39] C. L. Barrett, H. B. Hunt III, M. V. Marathe et al., "Predecessor and permutation existence problems for sequential dynamical systems," *Discrete Mathematics & Theoretical Computer Science AB(DMCS)*, pp. 69–80, 2003.

[40] W. Y. C. Chen, X. Li, and J. Zheng, "Matrix method for linear sequential dynamical systems on digraphs," *Applied Mathematics and Computation*, vol. 160, no. 1, pp. 197–212, 2005.

[41] H. S. Mortveit and C. M. Reidys, "Discrete, sequential dynamical systems," *Discrete Mathematics*, vol. 226, no. 1–3, pp. 281–295, 2001.

[42] S. Wolfram, "Statistical mechanics of cellular automata," *Reviews of Modern Physics*, vol. 55, no. 3, pp. 601–644, 1983.

[43] C. Barrett, H. B. Hunt III, M. V. Marathe et al., "Predecessor existence problems for finite discrete dynamical systems," *Theoretical Computer Science*, vol. 386, no. 1-2, pp. 3–37, 2007.

[44] C. Barrett, H. B. Hunt III, M. V. Marathe et al., "predecessor existence problems for finite discrete dynamical systems" [theoret. comput. sci. 386 (1-2) (2007) 3-37]," *Theoretical Computer Science*, vol. 395, no. 1, pp. 132-133, 2008.

Hindawi
Complexity
Volume 2019, Article ID 9261793, 9 pages
https://doi.org/10.1155/2019/9261793

WILEY | Hindawi

Research Article

Design of Fixed Points in Boolean Networks Using Feedback Vertex Sets and Model Reduction

Koichi Kobayashi

Graduate School of Information Science and Technology, Hokkaido University, Sapporo 060-0814, Japan

Correspondence should be addressed to Koichi Kobayashi; k-kobaya@ssi.ist.hokudai.ac.jp

Received 16 August 2018; Revised 8 November 2018; Accepted 13 February 2019; Published 4 March 2019

Guest Editor: Henning S. Mortveit

Fixed points in Boolean networks (BNs) represent cell types or states of cells and are important to decide characteristics of cells. As the control problem on fixed points, it is important to consider the problem of changing fixed points by using external stimuli (i.e., control inputs). In this paper, we propose two methods for designing fixed points. First, a design method using model reduction is proposed. Using the reduced model, the problem of placing fixed points can be rewritten as an integer linear programming problem. Next, we consider the design problem using only the graph structure of a given BN and derive some results. In both methods, a feedback vertex set of a directed graph plays an important role. Finally, a biological example is presented.

1. Introduction

One of the central problems in systems biology is to develop control theory of gene regulatory networks. Control of gene regulatory networks is closely related to therapeutic interventions, which are realized by radiation, chemotherapy, and so on. Towards gene therapy technologies in the future (see, e.g., [1]), establishment of theoretical foundations on modeling, analysis, and control of gene regulatory networks is important. Furthermore, experimental studies on control of gene regulatory networks have been done (see, e.g., [2–5]). Thus, much attention in control of gene regulatory networks has been attracted from both theoretical and practical viewpoints.

In control methods, a mathematical model should be chosen depending on characteristics of dynamics. A gene regulatory network is modeled by discrete dynamical systems (e.g., Boolean networks (BNs) [6]), continuous dynamical systems (e.g., differential/difference equations), or hybrid dynamical systems (e.g., piecewise affine models) (see, e.g., [7] for further details). In the last decade, BNs are widely used as a mathematical model for control methods of gene regulatory networks (see, e.g., [8–13]). In a BN, gene expression is modeled by a binary value (0 or 1), and interactions between genes are modeled by a set of Boolean functions.

While this model is quite simple, it is still useful in developing a control method for gene regulatory networks. A BN has been extended to a probabilistic BN (PBN) and a context-sensitive PBN (see, e.g., [14–17]).

In this paper, we focus on the control problem related to fixed points. Fixed points in BNs represent cell types or states of cells [18] and are important to decide characteristics of cells. Many results on analysis of fixed points have been obtained (see, e.g., [9, 19–22]). Change of fixed points by external stimuli has been analyzed in, e.g., [23]. However, to the best of our knowledge, the problem of designing fixed points by control has not been formally formulated and studied.

Furthermore, the semitensor product method has been widely used in analysis and control of BNs (see, e.g., [10, 11]). This method is powerful, but there is a serious technical issue. That is, matrices with at least $2^n \times 2^n$ size must be manipulated. Hence, it is important to develop the methods for simplifying a given BN.

In this paper, the problem of designing fixed points by external stimuli is studied, and two design methods are proposed. See also Figure 1 for an overview of the proposed methods.

First, the design method using model reduction is proposed. The model reduction method used in this paper is

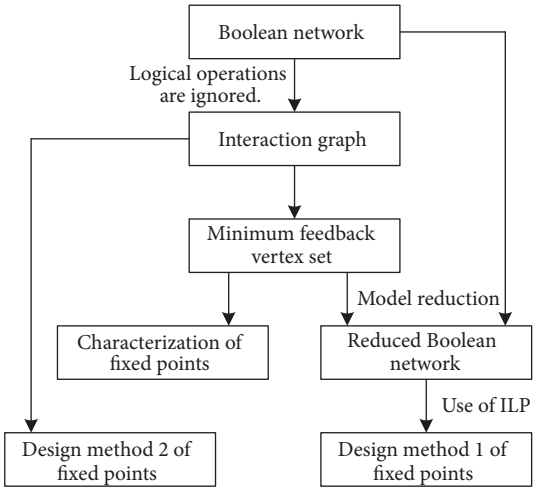


Figure 1: Overview of the proposed methods.

based on the methods in [24, 25]. Based on these methods, we propose a more sophisticated procedure using a minimum feedback vertex set of an interaction graph [20] obtained from a given BN. A feedback vertex set of a graph is a set of vertices whose removal results an acyclic graph. If the number of its elements is minimal, then it is called a minimum feedback vertex set. A feedback vertex set is well known as an important element to characterize the behavior of biological systems modeled by differential equations [26, 27]. In this paper, it is shown that a feedback vertex set is significant in model reduction of BNs. Using the reduced BN, the problem of designing fixed points by external stimuli can be rewritten as an integer linear programming (ILP) problem (see also Figure 1).

Next, the design method using an interaction graph obtained from a given BN is proposed. In this method, Boolean functions are not used, and only graph structure of a given BN is used. First, the number of fixed points is characterized by using a minimum feedback vertex set (see also Figure 1). Based on this result, a class of BNs such that the number of fixed points becomes 1 by external stimuli is clarified. Next, the design procedure using an interaction graph is proposed (see also Figure 1).

This paper is organized as follows. In Section 2, properties of BNs are explained. In Section 3, the design method using model reduction is explained. In Section 4, the design method using only an interaction graph is explained. In Section 5, a biological example on an apoptosis network is presented. In Section 6, we conclude this paper.

Notation. For the numbers $x_1, x_2, \ldots, x_n$ and the index set $\mathcal{I} = \{i_1, i_2, \ldots, i_m\} \subseteq \{1, 2, \ldots, n\}$, define $[x_i]_{i\in\mathcal{I}} := \begin{bmatrix} x_{i_1} & x_{i_2} & \cdots & x_{i_m} \end{bmatrix}^\top$. Let I_n and $0_{m\times n}$ denote the $n \times n$ identity matrix and the $m \times n$ zero matrix, respectively. For simplicity of notation, we sometimes use the symbol 0 instead of $0_{m\times n}$, and the symbol I instead of I_n.

2. Preliminaries

In this section, first, the outline of BNs is explained. Next, some definitions are given.

2.1. Boolean Networks. Consider the following BN:

$$\begin{aligned} x_1(k+1) &= f_1\left(\left[x_j(k)\right]_{j\in\mathcal{P}_1}, \left[\neg x_j(k)\right]_{j\in\mathcal{N}_1}\right), \\ x_2(k+1) &= f_2\left(\left[x_j(k)\right]_{j\in\mathcal{P}_2}, \left[\neg x_j(k)\right]_{j\in\mathcal{N}_2}\right), \\ &\vdots \\ x_n(k+1) &= f_n\left(\left[x_j(k)\right]_{j\in\mathcal{P}_n}, \left[\neg x_j(k)\right]_{j\in\mathcal{N}_n}\right), \end{aligned} \tag{1}$$

where $x_i \in \{0, 1\}$, $i \in \{1, 2, \ldots, n\}$ is the state (e.g., the expression of genes) and $k \in \{0, 1, 2, \ldots\}$ is the discrete time. The set $\mathcal{P}_i \subseteq \{1, 2, \ldots, n\}$ is a given index set of nodes that affecting the state x_i positively. The set $\mathcal{N}_i \subseteq \{1, 2, \ldots, n\} \setminus \mathcal{P}_i$ is a given index set of nodes affecting the state x_i negatively. Since $\mathcal{P}_i \cap \mathcal{N}_i = \emptyset$ holds, we consider only unate functions. The function $f_i : \{0, 1\}^{|\mathcal{P}_i|} \times \{0, 1\}^{|\mathcal{N}_i|} \longrightarrow \{0, 1\}^1$ is a given Boolean function satisfying the following assumptions:

(i) f_i is minimal (i.e., redundant terms such as $x_i \vee \neg x_i (= 1)$ are not included),

(ii) Logical operators in f_i are composed of AND ($\wedge$) and OR ($\vee$),

(iii) f_i is identical 0 or 1 if $\mathcal{P}_i = \mathcal{N}_i = \emptyset$ holds.

We present a simple example.

Example 1. Consider the following BN of an apoptosis network:

$$\begin{aligned} x_1(k+1) &= x_1(k), \\ x_2(k+1) &= x_1(k) \wedge \neg x_3(k), \\ x_3(k+1) &= \neg x_2(k) \wedge x_4(k), \\ x_4(k+1) &= x_1(k) \vee x_3(k), \end{aligned} \tag{2}$$

where we have the following:

x_1: the concentration level (high or low) of the tumor necrosis factor (TNF, a stimulus)

x_2: the concentration level of the inhibitor of apoptosis proteins (IAP)

x_3: the concentration level of the active caspase 3 (C3a)

x_4: the concentration level of the active caspase 8 (C8a)

This model is described in [28] and is a simplified version of an apoptosis network model in [23]. In this model, $x_3(k) = 0$ implies cell survival, and $x_2(k) = 0, x_3(k) = 1$ imply cell death [28]. In this BN, the following relations hold:

$$\begin{aligned} &\mathcal{P}_1 = \{1\}, \ \mathcal{N}_1 = \emptyset, \\ &\mathcal{P}_2 = \{1\}, \ \mathcal{N}_2 = \{3\}, \\ &\mathcal{P}_3 = \{4\}, \ \mathcal{N}_3 = \{2\}, \\ &\mathcal{P}_4 = \{1, 3\}, \ \mathcal{N}_3 = \emptyset. \end{aligned}$$

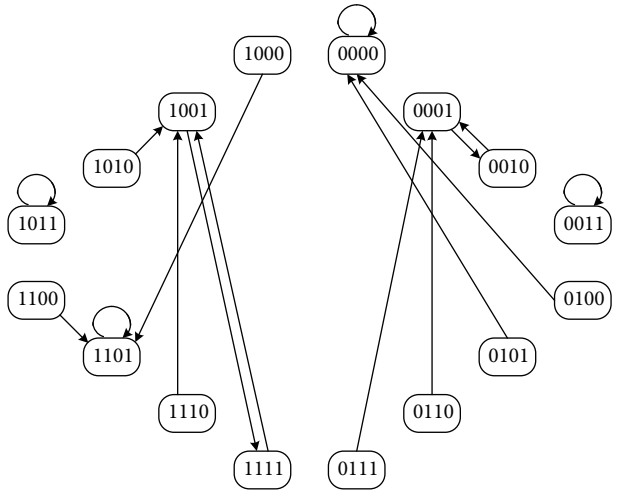


Figure 2: State transition diagram, where four fixed points and two cyclic attractors exist.

2.2. Definitions. Next, some definitions are given. A fixed point and a cyclic attractor are defined as follows.

Definition 2. The state $x(k)$ is called a *fixed point* if $x(k+1) = x(k)$ holds.

A fixed point is also called a *singleton attractor*.

Definition 3. A set of states with p elements, i.e., $\{x(k), x(k+1), \ldots, x(k+p-1)\}$ $(p \geq 2)$, is called a *cyclic attractor* if $x(k+p) = x(k)$ holds for all k.

We present a simple example.

Example 4. Consider the BN in Example 1 again. Then, the state transition diagram can be obtained as Figure 2, where each node corresponds to the value of the state. From this figure, we see that there exist four fixed points ($[0\ 0\ 0\ 0]^\top$, $[0\ 0\ 1\ 1]^\top$, $[1\ 0\ 1\ 1]^\top$, and $[1\ 1\ 0\ 1]^\top$) and two cyclic attractors ($\{[0\ 0\ 0\ 1]^\top, [0\ 0\ 1\ 0]^\top\}$ and $\{[1\ 0\ 0\ 1]^\top, [1\ 1\ 1\ 1]^\top\}$).

An interaction graph obtained from a given BN is defined as follows (see, e.g., [20]).

Definition 5. An interaction graph of BNs is defined by a signed directed graph $G = (\mathscr{V}, \mathscr{E}, L)$, where $\mathscr{V} = \{1, 2, \ldots, n\}$ is the set of vertices corresponding to x_i, $i \in \{1, 2, \ldots, n\}$, $\mathscr{E} = \{(j, i) \in \{1, 2, \ldots, n\} \times \{1, 2, \ldots, n\} \mid j \in \mathscr{P}_i \cup \mathscr{N}_i\}$ is the set of arcs, and $L : \mathscr{E} \longrightarrow \{+, -\}$ is the labeling function such that $L(j, i) = +$ if $j \in \mathscr{P}_i$ while $L(j, i) = -$ if $j \in \mathscr{N}_i$.

In the next example, an interaction graph is explained.

Example 6. Consider the BN in Example 1 again. Then, the interaction graph can be obtained as Figure 3. From this graph, we see that an interaction graph of a given BN can model interactions between genes.

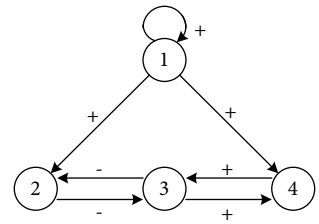


Figure 3: Interaction graph.

For a given interaction graph, a feedback vertex set is defined as follows (see, e.g., [29, 30]).

Definition 7. A set of vertices of an interaction graph is called a *feedback vertex set* if removal of vertices results in an acyclic graph. In particular, a feedback vertex set is called a *minimum feedback vertex set* if the number of its elements is minimum.

We remark here that, in the above definition, the sign (+ or −) in an interaction graph is ignored. The computational complexity of finding a minimum feedback vertex set is NP-complete [30], but an approximate algorithm of finding it has been developed (see, e.g., [29]).

Finally, in this paper, input vertices and output vertices for a given interaction graph are newly defined as follows.

Definition 8. A vertex of a given interaction graph is called an *input vertex* if it corresponds to the Boolean function f_i satisfying either $x_i(k+1) = x_i(k)$ (i.e., $\mathscr{P}_i = \{i\}$) or $x_i(k+1) = 0(1)$ (i.e., $\mathscr{P}_i = \emptyset$). In both cases, $\mathscr{N}_i = \emptyset$ holds.

In the case of $x_i(k+1) = x_i(k)$, if $x_i(0)$ can be arbitrarily determined, then $x_i(k)$ can be regarded as a constant input. In the case of $x_i(k+1) = 0(1)$, even if $x_i(0)$ can be arbitrarily determined, $x_i(k)$, $k \geq 1$ cannot be controlled. In gene regulatory networks, an input vertex corresponds to an external stimulus from out of the cell. In other words, the state corresponding to the input vertex is not influenced by other states. For simplicity, we may consider that the state corresponding to the input vertex is regarded as a constant.

Definition 9. A vertex of a given interaction graph is called an *output vertex* if it corresponds to the Boolean function f_i satisfying $\mathscr{P}_j \cap \{i\} = \emptyset$, $\mathscr{N}_j \cap \{i\} = \emptyset$, $j \neq i$.

In other words, the state corresponding to the output vertex does not influence other states.

We present a simple example.

Example 10. Consider the interaction graph in Figure 3. The minimum vertex set is given by $\{1, 3\}$. The input vertex is given by node 1. In this interaction graph, there is no output vertex.

Hereafter, the following assumption is made for input vertices.

Assumption 11. There exists no input vertex such that the Boolean function is given by $x_i(k+1) = 0(1)$. In addition,

for the state corresponding to the input vertex, its initial state can be arbitrarily controlled.

In other words, initial states corresponding to input vertices are regarded as a control input.

Example 12. Consider the BN in Example 1 again. The input vertex is given by 1 (see also Figure 3). The state transition diagram of this BN is shown in Figure 2. From Figure 2, we see that fixed points are changed depending on the value of $x_1(0)$. In the case of $x_1(0) = 1$ (i.e., $x_1(k) = 0$ for $k \geq 1$), fixed points are given by $\begin{bmatrix}1 & 0 & 1 & 1\end{bmatrix}^\top$ and $\begin{bmatrix}1 & 1 & 0 & 1\end{bmatrix}^\top$. In the case of $x_1(0) = 0$, fixed points are given by $\begin{bmatrix}0 & 0 & 0 & 0\end{bmatrix}^\top$ and $\begin{bmatrix}0 & 0 & 1 & 1\end{bmatrix}^\top$.

Hereafter, under Assumption 1l, we consider the problem of designing fixed points using initial values of states corresponding to input vertices. Input vertices correspond to external stimuli from out of cells (see, e.g., Example 1 and Section 5). Furthermore, it is difficult to change the values of external stimuli dynamically. Hence, it is appropriate to consider the design problem under Assumption 1l.

3. Design of Fixed Points Using Model Reduction

In this section, we consider designing fixed points using reduced BNs. First, a model reduction method for BNs is explained. Model reduction of BNs has been studied so far (see, e.g., [24, 25]). In this paper, using these results and the notion of minimum feedback vertex sets, we introduce a sophisticated procedure of model reduction. In the procedure below, the number of states in the reduced model can be determined at the first step. Hence, this procedure is visible and simple. Next, we consider the problem of designing fixed points using reduced BNs.

3.1. Model Reduction of Boolean Networks. Let $\mathscr{W}$ denote a set of vertices consisting of a minimum feedback vertex set and a set of output vertices. The procedure of model reduction introduced in this paper is given as follows.

Procedure of Model Reduction of BNs

Step 1. For a given interaction graph, find a minimum feedback vertex set and a set of output vertices. Let $\widehat{x}_i$ and $\widehat{f}_i(\cdot)$, $i \in \mathscr{W}$ denote the state and the Boolean function corresponding to vertices obtained, respectively.

Step 2. Replace $x_j(k+1) = f_j(\cdot)$, $j \in \{1, 2, \ldots, n\} \setminus \mathscr{W}$ with $x_j(k) = f_j(\cdot)$.

Step 3. Replace $x_j(k)$ appearing in $\widehat{f}_i(\cdot)$ with $f_j(\cdot)$. Repeat until there exists no $x_j(k)$ appearing in $\widehat{f}_i(\cdot)$.

Step 4. Simplify the Boolean functions $\widehat{f}_i$, $i \in \mathscr{W}$.

We remark here that, under Assumption 1l, all input vertices are included in a minimum feedback vertex set. In addition, from the definition of minimum feedback vertex sets, it is guaranteed that Step 3 is terminated. We explain this procedure using a simple example.

Example 13. The BN in Example 1 is considered. In Step 1, we can obtain $\mathscr{W} = \{1, 3\}$. In Step 2, we can obtain $x_2(k) = \widehat{x}_1(k) \wedge \neg\widehat{x}_3(k)$ and $x_4(k) = \widehat{x}_1(k) \vee \widehat{x}_3(k)$. In Step 3, we can obtain

$$\begin{aligned}\widehat{x}_1(k+1) &= \widehat{x}_1(k),\\ \widehat{x}_3(k+1) &= \neg\left(\widehat{x}_1(k) \wedge \neg\widehat{x}_3(k)\right) \wedge \left(\widehat{x}_1(k) \vee \widehat{x}_3(k)\right).\end{aligned} \tag{3}$$

Finally, in Step 4, we can obtain

$$\begin{aligned}\widehat{x}_1(k+1) &= \widehat{x}_1(k),\\ \widehat{x}_3(k+1) &= \widehat{x}_3(k).\end{aligned} \tag{4}$$

Hereafter, the reduced model obtained by the above procedure is denoted by

$$\begin{aligned}\widehat{x}_1(k+1) &= \widehat{f}_1\left(\left[\widehat{x}_j(k)\right]_{j\in\widehat{\mathscr{P}}_1}, \left[\neg\widehat{x}_j(k)\right]_{j\in\widehat{\mathscr{N}}_1}\right),\\ \widehat{x}_2(k+1) &= \widehat{f}_2\left(\left[\widehat{x}_j(k)\right]_{j\in\widehat{\mathscr{P}}_2}, \left[\neg\widehat{x}_j(k)\right]_{j\in\widehat{\mathscr{N}}_2}\right),\\ &\vdots\\ \widehat{x}_{\widehat{n}}(k+1) &= \widehat{f}_{\widehat{n}}\left(\left[\widehat{x}_j(k)\right]_{j\in\widehat{\mathscr{P}}_{\widehat{n}}}, \left[\neg\widehat{x}_j(k)\right]_{j\in\widehat{\mathscr{N}}_{\widehat{n}}}\right),\end{aligned} \tag{5}$$

where $\widehat{n} := |\mathscr{W}|$.

For the reduced model obtained, the following theorem has been obtained [25].

Theorem 14. *The set of fixed points for the BN (1) and the set of fixed points for the reduced BN (5) are one-to-one correspondence.*

We present a simple example.

Example 15. Consider the BN in Example 1 again. From Figure 2, we see that there exist four fixed points. On the other hand, the reduced model for (2) is given by (4). From (4), we see that there exist four fixed points ($\begin{bmatrix}0 & 0\end{bmatrix}^\top$, $\begin{bmatrix}0 & 1\end{bmatrix}^\top$, $\begin{bmatrix}1 & 0\end{bmatrix}^\top$, and $\begin{bmatrix}1 & 1\end{bmatrix}^\top$). Thus, we see that fixed points for (2) and fixed points for (4) are one-to-one correspondence.

Furthermore, from the above procedure and the definitions of feedback vertex sets and output vertices, we can obtain the following theorem immediately.

Theorem 16. *The minimal number of states for the reduced BN is given by a sum of the number of elements in a minimum feedback vertex set and the number of output vertices.*

If it is difficult to find a minimum feedback vertex set; then instead of it, we may use a feedback vertex set. In such a case, the minimality of the reduced model is lost. However, in large-scale networks, there are situations that the reduced model with no minimality is useful.

3.2. Design of Fixed Points Using Reduced Boolean Networks. Using the reduced BN, consider designing fixed points. It is difficult to uniquely determine a fixed point. We consider the following problem.

Problem 17. Consider the reduced BN (5). Suppose that for states corresponding to vertices except for input vertices; desired fixed points $\alpha_1, \alpha_2, \ldots, \alpha_{n_d} \in \{0,1\}^{\widehat{n}-m}$ and undesired fixed points $\beta_1, \beta_2, \ldots, \beta_{n_u} \in \{0,1\}^{\widehat{n}-m}$ are given, where m is the number of input vertices. Find initial values of the states corresponding to the input vertices such that the reduced BN has desired fixed points and has no undesired fixed points.

Fixed points represent cell types or states of cells [18]. Hence, the above problem is important to characterize the property of cells. Since, in this problem, we focus on only fixed points, we can use the reduced BN.

This problem can be equivalently rewritten as an integer linear programming (ILP) problem. We explain this fact below. First, as a preparation, the following lemma [31] is introduced.

Lemma 18. *Consider two binary variables δ_1 and δ_2. Then, the following relations hold:*

(i) $\neg\delta_1$ is equivalent to $1-\delta_1$.

(ii) $\delta_1 \wedge \delta_2$ is equivalent to $\delta_1\delta_2$.

(iii) $\delta_1 \vee \delta_2$ is equivalent to $\delta_1 + \delta_2 - \delta_1\delta_2$.

Using this lemma, (5) can be equivalently rewritten as a polynomial system with binary variables.

Furthermore, the following lemma [32] is also introduced.

Lemma 19. *Suppose that binary variables $\delta_j \in \{0,1\}$, $j \in \mathcal{J}$ are given, where $\mathcal{J}$ is some index set. Then $z = \prod_{j\in\mathcal{J}}\delta_j$ is equivalent to the following linear inequalities:*

$$\begin{aligned} \sum_{j\in\mathcal{J}}\delta_j - z &\le |\mathcal{J}| - 1, \\ -\sum_{j\in\mathcal{J}}\delta_j + |\mathcal{J}|\, z &\le 0, \end{aligned} \tag{6}$$

where $|\mathcal{J}|$ is the cardinality of $\mathcal{J}$.

Using Lemmas 18 and 19, the reduced BN (5) can be equivalently rewritten as the following pair of a linear state equation and a linear inequality:

$$\begin{aligned} \begin{bmatrix} \widehat{x}(k+1) \\ u(k+1) \end{bmatrix} &= \begin{bmatrix} A_1 & A_2 \\ 0 & I \end{bmatrix} \begin{bmatrix} \widehat{x}(k) \\ u(k) \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} z(k), \\ C \begin{bmatrix} \widehat{x}(k) \\ u(k) \end{bmatrix} &+ Dz(k) \le E, \end{aligned} \tag{7}$$

where $\widehat{x} \in \{0,1\}^{\widehat{n}-m}$ is the vector consisting of states corresponding to vertices except for input vertices, and $u \in \{0,1\}^m$ consists of states corresponding to input vertices. The vector $z \in \{0,1\}^p$ is the auxiliary binary variable, and the dimension p can be determined from Boolean functions. See Section 5 for a method to derive (7). See also [33]. Then, we can obtain the following lemma.

Lemma 20. *Problem 17 is equivalent to the following problem.*

Problem 21. Find $u \in \{0,1\}^m$, $z_1^d, z_2^d, \ldots, z_{n_d}^d \in \{0,1\}^p$, and $z_1^u, z_2^u, \ldots, z_{n_u}^u \in \{0,1\}^p$ subject to

$$\alpha_i = A_1\alpha_i + A_2u + Bz_i^d, \quad i \in \{1,2,\ldots,n_d\}, \tag{8}$$

$$C\begin{bmatrix} \alpha_i \\ u \end{bmatrix} + Dz_i^d \le E, \quad i \in \{1,2,\ldots,n_d\} \tag{9}$$

and

$$\beta_i \ne A_1\beta_i + A_2u + Bz_i^u, \quad i \in \{1,2,\ldots,n_u\}, \tag{10}$$

$$C\begin{bmatrix} \beta_i \\ u \end{bmatrix} + Dz_i^u \le E, \quad i \in \{1,2,\ldots,n_u\}. \tag{11}$$

Finally, consider (10). Noting that the left-hand side value and the right-hand side value are binary, (10) is equivalent to

$$(A_1 - I)\beta_i + A_2u + Bz_i^u = \delta_1 - \delta_2, \tag{12}$$

$$\delta_1 + \delta_2 = 1, \tag{13}$$

where $\delta_1, \delta_2 \in \{0,1\}$. Thus, we can obtain the following theorem.

Theorem 22. *Problem 17 is equivalent to the following ILP problem.*

Problem 23. Find $u \in \{0,1\}^m$, $z_1^d, z_2^d, \ldots, z_{n_d}^d \in \{0,1\}^p$, $z_1^u, z_2^u, \ldots, z_{n_u}^u \in \{0,1\}^p$, and $\delta_1, \delta_2 \in \{0,1\}$ subject to (8), (9), (11), (12), and (13).

An ILP problem can be solved by using a free/commercial solver. In this paper, Problem 17 is transformed into an ILP problem, but as another approach, we may utilize an SMT (Satisfiability Modulo Theories) solver such as the Yices SMT solver [34].

4. Design of Fixed Points Using Minimum Feedback Vertex Sets and Interaction Graphs

In the previous subsection, we use Boolean functions of the reduced BN (5). However, there is a possibility that, even if a given BN is reduced, then solving the ILP problem (Problem 23) is hard. For such cases, structural analysis using only the interaction graph of a given BN is effective. In this section, first, we consider characterizing the number of fixed points using minimum feedback vertex sets. Next, we consider a general case using interaction graphs.

4.1. Special Cases. Consider two special cases on a minimum feedback vertex set. First, we can obtain the following theorem.

Theorem 24. *Consider the BN (1). Assume that a minimum feedback vertex set is empty. Then, the number of fixed points is 1.*

Proof. From the assumption, the states in the Boolean functions in (5) consist of the states corresponding to the output vertices. Since a minimum feedback vertex set is empty, the output vertex has no self-loop. Hence, the Boolean functions in the reduced BN (5) are given by either $\widehat{x}_i(k+1) = 0$ or $\widehat{x}_i(k+1) = 1$; that is, the fixed point is uniquely determined. □

The assumption in this theorem is equivalent to that a given interaction graph is acyclic. The same result has been obtained in [9, 20], but it is here reformulated using a minimum feedback vertex set.

Next, we consider the case where a minimum feedback vertex set is not empty.

Theorem 25. *Consider the BN (1). Assume that all elements of a minimum feedback vertex set are given by only input vertices. Then, the fixed point is uniquely determined by specifying the initial values of the states corresponding to the input vertices.*

Proof. From the assumption, the Boolean functions in (5) are given by a function of the states corresponding to the input vertices. Furthermore, from Assumption 1l, the Boolean functions corresponding to input vertices are given by the form of $\widehat{x}_i(k+1) = \widehat{x}_i(k)$. Hence, Theorem 25 can be obtained. □

From this theorem, we see that in the case where a minimum feedback vertex set is given by a set of input vertices; the fixed point can be controlled by the initial values of the states corresponding to the input vertices.

We present a simple example.

Example 26. Consider the following BN:

$$\begin{aligned} x_1(k+1) &= x_1(k), \\ x_2(k+1) &= \neg x_1(k), \\ x_3(k+1) &= x_2(k) \wedge x_4(k), \\ x_4(k+1) &= x_1(k). \end{aligned} \tag{14}$$

In the interaction graph of this BN, the input vertex is only 1. The fixed points are derived as $\begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix}^\top$ and $\begin{bmatrix} 1 & 0 & 0 & 1 \end{bmatrix}^\top$. Hence, the fixed points can be uniquely specified by using $x_1(0)$.

4.2. General Case. For two special cases, the number of fixed points was characterized. However, the concrete values of fixed points are not discussed. Here, we consider deriving a solution of the following problem.

Problem 27. Consider the BN (1). Suppose that, for states corresponding to vertices except for input vertices, a desired fixed point $\alpha \in \{0,1\}^{n-m}$ is given, where m is the number of input vertices. Then, using only the interaction graph of the BN, find initial values of the states corresponding to the input vertices such that the BN has a desired fixed point.

In general, it is necessary to determine α from the biological viewpoint. Hereafter, for simplicity of notation, states corresponding to vertices except for input vertices are given by $x_1, x_2, \ldots, x_{n-m}$, and states corresponding to input vertices are given by $x_{n-m+1}, x_{n-m+2}, \ldots, x_n$. Let α_i denote the i-th element of α.

The design procedure is given as follows.

Procedure of Designing Initial Values of States Corresponding to Input Vertices from an Interaction Graph

Step 1. Enumerate all combinations of initial values of $\begin{bmatrix} x_{n-m+1}(k) & x_{n-m+2}(k) & \cdots & x_n(k) \end{bmatrix}^\top$. Let γ_i, $i \in \{1, 2, \ldots, 2^m\}$ denote combinations of initial values obtained.

Step 2. Set $i = 1$.

Step 3. Substitute α and γ_i into the BN (1).

Step 4. If all arguments of f_i are equal to α_i, then a solution of Problem 27 is obtained as γ_i, and the procedure is terminated. If this condition does not hold, and $i < 2^m$, then set $i := i + 1$ and go to Step 3. If this condition does not hold, and $i = 2^m$, then a solution of Problem 27 cannot be found, and the procedure is terminated.

We remark here that, in this procedure, Boolean functions are not computed, and only information obtained from an interaction graph is used.

We present a simple example.

Example 28. Consider the following BN:

$$\begin{aligned} x_1(k+1) &= \neg x_2(k) \wedge x_4(k), \\ x_2(k+1) &= \neg x_1(k) \wedge x_2(k), \\ x_3(k+1) &= x_2(k) \vee x_5(k), \\ x_4(k+1) &= x_4(k), \\ x_5(k+1) &= x_5(k). \end{aligned} \tag{15}$$

Since Boolean functions are not explicitly used in this method, this BN is denoted by

$$\begin{aligned} x_1(k+1) &= f_1\left(\neg x_2(k), x_4(k)\right), \\ x_2(k+1) &= f_2\left(\neg x_1(k), x_2(k)\right), \\ x_3(k+1) &= f_3\left(x_2(k), x_5(k)\right), \end{aligned} \tag{16}$$

where $x_4(k+1) = x_4(k)$ and $x_5(k+1) = x_5(k)$ are omitted. In the interaction graph of this BN, the input vertices are 4 and 5. Suppose that α is given by $\alpha = \begin{bmatrix} 0 & 1 & 1 \end{bmatrix}^\top$. First, consider the

case of $x_4(0) = x_5(0) = 0$. In Step 4 of the above procedure, we can obtain

$$\begin{aligned} 0 &= f_1(0,0), \\ 1 &= f_2(1,1), \\ 1 &= f_3(1,0). \end{aligned} \tag{17}$$

Hence, the condition of Step 4 is not satisfied. Next, consider the case of $x_4(0) = 0$ and $x_5(0) = 1$. In Step 4, we can obtain

$$\begin{aligned} 0 &= f_1(0,0), \\ 1 &= f_2(1,1), \\ 1 &= f_3(1,1). \end{aligned} \tag{18}$$

Hence, the condition of Step 4 is satisfied. Thus, we see that α becomes one of the fixed points by setting $x_4(0) = 0$ and $x_5(0) = 1$.

The number of combinations of x_4 and x_5 is four. For all combinations, α is one of the fixed points. However, the condition of Step 4 of the above procedure holds in only the case of $x_4(0) = 0$ and $x_5(0) = 1$. Since Boolean functions are not used, the above procedure provides us a sufficient condition such that α is one of the fixed points.

As the other example, suppose that α is given by $\alpha = [1\ 0\ 0]^\top$. In the case of $x_4(0) = 1$ and $x_5(0) = 0$, we can obtain

$$\begin{aligned} 1 &= f_1(1,1), \\ 0 &= f_2(0,0), \\ 0 &= f_3(0,0). \end{aligned} \tag{19}$$

We see that α becomes one of the fixed points by setting $x_4(0) = 1$ and $x_5(0) = 0$. The conditions $x_4(0) = 1$ and $x_5(0) = 0$ are the unique condition such that α is a fixed point.

Thus, the proposed procedure corresponds to a sufficient condition such that α is one of the fixed points, but computation of this procedure is easy. Since the proposed procedure is a sufficient condition, there is a possibility that even if α is one of the fixed points, the proposed procedure cannot be found this fact. It is one of the future efforts to derive a better sufficient condition based on structural analysis.

5. Biological Example

In Example 1, we used a very simple model of an apoptosis network. In this section, a more detailed model [23] is used.

Consider the following BN expressing an apoptosis network (see also Figure 4):

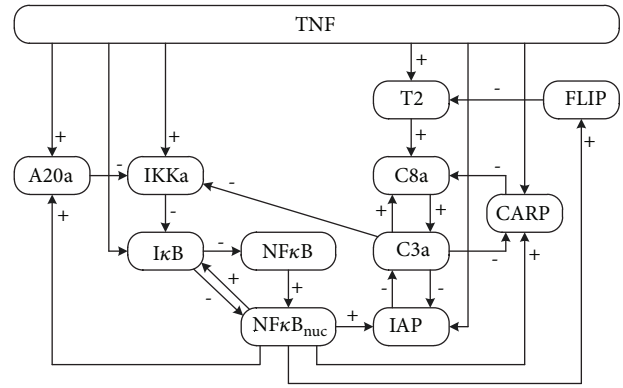


Figure 4: Interaction graph of an apoptosis network addressed in [23]. Some edges that do not have a sign imply that both signs are assigned.

$$\begin{aligned} x_1(k+1) &= \neg x_8(k) \wedge x_{12}(k), \\ x_2(k+1) &= \neg x_6(k) \wedge \neg x_9(k) \wedge x_{12}(k), \\ x_3(k+1) &= \neg x_5(k), \\ x_4(k+1) &= x_3(k) \wedge \neg x_5(k), \\ x_5(k+1) &= \left(\neg x_2(k) \wedge x_4(k) \wedge x_{12}(k)\right) \\ &\quad \vee \left\{\left(\neg x_2(k) \vee x_4(k)\right) \wedge \neg x_{12}(k)\right\}, \\ x_6(k+1) &= x_4(k) \wedge x_{12}(k), \\ x_7(k+1) &= \left(x_4(k) \wedge \neg x_9(k) \wedge x_{12}(k)\right) \\ &\quad \vee \left\{\left(x_4(k) \vee \neg x_9(k)\right) \wedge \neg x_{12}(k)\right\}, \\ x_8(k+1) &= x_4(k), \\ x_9(k+1) &= \neg x_7(k) \wedge x_{10}(k), \\ x_{10}(k+1) &= \left(x_1(k) \vee x_9(k)\right) \wedge \neg x_{11}(k) \\ x_{11}(k+1) &= \left(x_4(k) \wedge \neg x_9(k) \wedge x_{12}(k)\right), \\ &\quad \vee \left\{\left(x_4(k) \vee \neg x_9(k)\right) \wedge \neg x_{12}(k)\right\}, \\ x_{12}(k+1) &= x_{12}(k), \end{aligned} \tag{20}$$

where $x_1, x_2, \ldots, x_{12}$ are as follows:

x_1: a second complex (T2),

x_2: inhibitor of IκB kinases (IKKa),

x_3: nuclear factor κB in the cytoplam (NFκB),

x_4: nuclear factor κB in the nucleus (NFκB$_{nuc}$),

x_5: inhibitor of NFκB (IκB),

x_6: a protein regulating IKK activity (A20a),

x_7: inhibitor of apoptosis proteins (IAP),

x_8: a protein associated with inhibitation of complex T2 (FLIP),

x_9: active caspase 3 (C3a),

x_{10}: active caspase 8 (C8a),

x_{11}: caspase-8 and -10-associated RING proteins (CARP),

x_{12}: tumor necrosis factor (TNF).

The state corresponding to the input vertex is given by x_1. Furthermore, in this model, $x_7 = 0$ and $x_9 = 1$ imply cell death, and $x_7 = 1$ and $x_9 = 0$ imply cell survival. This BN has two fixed points, i.e., $[0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 1\ 0\ 0]^\top$ and $[0\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 0\ 0\ 1\ 0]^\top$, and has no fixed point for which $x_{12} = 1$ holds. We remark that in [23], the asynchronous BN dynamics are considered, and its fixed points are different to those in the synchronous BN dynamics studied in this paper. Based on observations made for the simplified model in Example 1, we next consider modifying the dynamics of x_5 by $x_5(k+1) = (\neg x_2(k) \vee x_4(k)) \wedge \neg x_{12}(k)$. The resulting BN has the four fixed points $[0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 1\ 0\ 0]^\top$, $[0\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 0\ 0\ 1\ 0]^\top$, $[0\ 0\ 1\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 0\ 1]^\top$, and $[0\ 0\ 1\ 1\ 0\ 1\ 1\ 1\ 0\ 0\ 1\ 1]^\top$.

Consider applying the method in Section 3 to this apoptosis model. From Figure 4, we see that one of the minimum vertex sets is given by $\{4, 9, 12\}$ (i.e., NFκB$_{\text{nuc}}$, C3a, and TNF). Then, we can obtain the following reduced BN:

$$\begin{aligned} \widehat{x}_4(k+1) &= \widehat{x}_{12}(k), \\ \widehat{x}_9(k+1) &= \{\neg \widehat{x}_4(k) \wedge (\widehat{x}_9(k) \vee \widehat{x}_{12}(k))\} \\ &\quad \vee (\widehat{x}_4(k) \wedge \widehat{x}_9(k) \wedge \widehat{x}_{12}(k)), \\ \widehat{x}_{12}(k+1) &= \widehat{x}_{12}(k). \end{aligned} \tag{21}$$

This reduced BN has the four fixed points $[0\ 0\ 0]^\top$, $[0\ 1\ 0]^\top$, $[1\ 0\ 1]^\top$, and $[1\ 1\ 1]^\top$, and these are in a one-to-one correspondence with the fixed points of the modified BN above.

According to [23], $\widehat{x}_{12}(k)$ (TNF) is regarded as a constant control input of the whole system. The above network is composed of a proapoptotic and an antiapoptotic pathway, which are activated by stimulation of death receptor by a factor such as TNF. Binding of TNF to a death receptor activates the antiapoptotic pathway and, after a certain delay, the proapoptotic pathway is also activated. In the above network, we cannot choose either cell death or cell survival by controlling only TNF. The choice of being either cell death or survival depends on also the initial value of other states. Since this BN is too simple, we can obtain the following facts from the truth table:

(i) By setting $\widehat{x}_{12}(0) = 0$, fixed points can be placed to $[\widehat{x}_4\ \ \widehat{x}_9]^\top = [0\ \ 0]^\top$ and $[\widehat{x}_4\ \ \widehat{x}_9]^\top = [0\ \ 1]^\top$.

(ii) By setting $\widehat{x}_{12}(0) = 1$, fixed points can be placed to $[\widehat{x}_4\ \ \widehat{x}_9]^\top = [1\ \ 0]^\top$ and $[\widehat{x}_4\ \ \widehat{x}_9]^\top = [1\ \ 1]^\top$.

Also from this result, we see that it is impossible to choose either cell death or cell survival by controlling only $\widehat{x}_{12}(0)$. These properties hold for the original BN. We suggest that additional control inputs are required.

Next, consider solving Problem 17 (i.e., the ILP problem). Here, we derive (7) under $u(k) = \widehat{x}_{12}(k)$. By a simple calculation, we can obtain

$$\begin{aligned} \begin{bmatrix} \widehat{x}_4(k+1) \\ \widehat{x}_9(k+1) \\ u(k+1) \end{bmatrix} &= \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \widehat{x}_4(k) \\ \widehat{x}_9(k) \\ u(k) \end{bmatrix} \\ &\quad + \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} z_1(k) \\ z_2(k) \\ z_3(k) \end{bmatrix}, \end{aligned} \tag{22}$$

where $z_1(k) = \widehat{x}_4(k)\widehat{x}_9(k)$, $z_2(k) = \widehat{x}_4(k)u(k)$, and $z_3(k) = \widehat{x}_9(k)u(k)$, which can be transformed into a set of linear inequalities with respect to $\widehat{x}_4(k)$, $\widehat{x}_9(k)$, $u(k)$ by using Lemma 19. In Problem 17, we set $\alpha_1 = [0\ \ 0]^\top$ (the desired fixed point) and $\beta_1 = [1\ \ 1]^\top$ (the undesired fixed point). This setting is artificially given. By solving the ILP problem (Problem 23), we can obtain $\widehat{x}_{12}(0) = 0$.

6. Conclusion

In this paper, we studied the design problem of fixed points in BNs. First, the design problem using model reduction was considered. This problem is rewritten as an ILP problem. Next, the design problem using only interaction graphs was considered. Finally, a biological example is presented.

There are several open problems.

In the proposed methods, a minimum feedback set of the interaction graph obtained from a given BN plays an important role. We will consider that this fact does not depend on mathematical models and is the design principle in biological systems. However, further discussion from several viewpoints is required.

Since we consider only a simple BN, one of the future efforts is to apply the proposed methods to several biological systems. Reducing conservativeness of the procedure in Section 4.2 using additional information is one of the future efforts. In this paper, we focus on only fixed points, and we do not consider cyclic attractors. Analysis of cyclic attractors is also one of the future efforts.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The author declares that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was partly supported by JSPS KAKENHI Grant no. 17K06486.

References

[1] H. Santos-Rosa and C. Caldas, "Chromatin modifier enzymes, the histone code and cancer," *European Journal of Cancer*, vol. 41, no. 16, pp. 2381–2402, 2005.

[2] F. Menolascina, G. Fiore, E. Orabona et al., "In-vivo real-time control of protein expression from endogenous and synthetic gene networks," *PLoS Computational Biology*, vol. 10, no. 5, pp. 1–14, 2014.

[3] A. Milias-Argeitis, S. Summers, J. Stewart-Ornstein et al., "In silico feedback for in vivo regulation of a gene expression circuit," *Nature Biotechnology*, vol. 29, no. 12, pp. 1114–1116, 2011.

[4] G. P. Pathak, J. D. Vrana, and C. L. Tucker, "Optogenetic control of cell function using engineered photoreceptors," *Biology of the Cell*, vol. 105, no. 2, pp. 59–72, 2013.

[5] J. Uhlendorf, A. Miermont, T. Delaveau et al., "Long-term model predictive control of gene expression at the population and single-cell levels," *Proceedings of the National Acadamy of Sciences of the United States of America*, vol. 109, no. 35, pp. 14271–14276, 2012.

[6] S. A. Kauffman, "Metabolic stability and epigenesis in randomly constructed genetic nets," *Journal of Theoretical Biology*, vol. 22, no. 3, pp. 437–467, 1969.

[7] H. de Jong, "Modeling and simulation of genetic regulatory systems: a literature review," *Journal of Computational Biology*, vol. 9, no. 1, pp. 67–103, 2002.

[8] T. Akutsu, M. Hayashida, W.-K. Ching, and M. K. Ng, "Control of Boolean networks: hardness results and algorithms for tree structured networks," *Journal of Theoretical Biology*, vol. 244, no. 4, pp. 670–679, 2007.

[9] S. Azuma, T. Yoshida, and T. Sugie, "Structural monostability of activation-inhibition Boolean networks," *IEEE Transactions on Control of Network Systems*, vol. 4, no. 2, pp. 179–190, 2017.

[10] D. Cheng and H. Qi, "Controllability and observability of Boolean control networks," *Automatica*, vol. 45, no. 7, pp. 1659–1667, 2009.

[11] D. Cheng, H. Qi, and Z. Li, *Analysis and Control of Boolean Network: A Semi-Tensor Product Approach*, Communications and Control Engineering Series, Springer, London, UK, 2011.

[12] K. Kobayashi, J. Imura, and K. Hiraishi, "Polynomial-time algorithm for controllability test of a class of boolean biological networks," *Eurasip Journal on Bioinformatics and Systems Biology*, vol. 2010, Article ID 210685, 12 pages, 2010.

[13] K. Kobayashi and K. Hiraishi, "Optimal control of gene regulatory networks with effectiveness of multiple drugs: a boolean network approach," *BioMed Research International*, vol. 2013, Article ID 246761, 11 pages, 2013.

[14] B. Faryabi, G. Vahedi, J.-F. Chamberland, A. Datta, and E. R. Dougherty, "Intervention in context-sensitive probabilistic boolean networks revisited," *EURASIP Journal on Bioinformatics and Systems Biology*, vol. 2009, Article ID 360864, 13 pages, 2009.

[15] K. Kobayashi and K. Hiraishi, "Optimization-based approaches to control of probabilistic Boolean networks," *Algorithms*, vol. 10, no. 1, article 13, 16 pages, 2017.

[16] R. Pal, A. Datta, M. L. Bittner, and E. R. Dougherty, "Intervention in context-sensitive probabilistic Boolean networks," *Bioinformatics*, vol. 21, no. 7, pp. 1211–1218, 2005.

[17] I. Shmulevich, E. R. Dougherty, S. Kim, and W. Zhang, "Probabilistic Boolean networks: a rule-based uncertainty model for gene regulatory networks," *Bioinformatics*, vol. 18, no. 2, pp. 261–274, 2002.

[18] S. A. Kauffman, *The Origins of Order: Self-organization and Selection in Evolution*, Oxford University Press, Oxford, UK, 1993.

[19] A. L. Coënt, L. Fribourg, and R. Soulat, "Compositional analysis of Boolean networks using local fixed-point iterations," in *Proceedings of the International Workshop on Reachability Problems*, pp. 134–147, Springer, Aalborg, Denmark, 2016.

[20] L. Paulevé and A. Richard, "Static analysis of Boolean networks based on interaction graphs: a survey," *Electronic Notes in Theoretical Computer Science*, vol. 284, pp. 93–104, 2012.

[21] T. Tamura and T. Akutsu, "Detecting a singleton attractor in a Boolean network utilizing sat algorithms," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E92-A, no. 2, pp. 493–501, 2009.

[22] A. Veliz-Cuba, B. Aguilar, F. Hinkelmann, and R. Laubenbacher, "Steady state analysis of Boolean molecular network models via model reduction and computational algebra," *BMC Bioinformatics*, vol. 15, no. 1, article no. 221, 2014.

[23] L. Tournier and M. Chaves, "Uncovering operational interactions in genetic networks using asynchronous Boolean dynamics," *Journal of Theoretical Biology*, vol. 260, no. 2, pp. 196–209, 2009.

[24] A. Naldi, E. Remy, D. Thieffry, and C. Chaouiya, "A reduction of logical regulatory graphs preserving essential dynamical properties," in *Proceedings of the International Conference on Computational Methods in Systems Biology*, pp. 266–280, Springer, Bologna, Italy, 2009.

[25] A. Veliz-Cuba, "Reduction of Boolean network models," *Journal of Theoretical Biology*, vol. 289, pp. 167–172, 2011.

[26] B. Fiedler, A. Mochizuki, G. Kurosawa, and D. Saito, "Dynamics and control at feedback vertex sets. I: informative and determining nodes in regulatory networks," *Journal of Dynamics and Differential Equations*, vol. 25, no. 3, pp. 563–604, 2013.

[27] A. Mochizuki, B. Fiedler, G. Kurosawa, and D. Saito, "Dynamics and control at feedback vertex sets. II: a faithful monitor to determine the diversity of molecular activities in regulatory networks," *Journal of Theoretical Biology*, vol. 335, pp. 130–146, 2013.

[28] M. Chaves, "Methods for qualitative analysis of genetic networks," in *Proceedings of 2009 European Control Conference (ECC)*, pp. 671–676, IEEE, Budapest, Hungary, August 2009.

[29] G. Even, J. Naor, B. Schieber, and M. Sudan, "Approximating minimum feedback sets and multicuts in directed graphs," *Algorithmica*, vol. 20, no. 2, pp. 151–174, 1998.

[30] R. M. Karp, "Reducibility among combinatorial problems," in *Complexity of Computer Computations*, pp. 85–103, Springer, New York, NY, USA, 1972.

[31] H. P. Williams, *Model Building in Mathematical Programming*, Willey, West Sussex, UK, 5th edition, 2013.

[32] T. M. Cavalier, P. M. Pardalos, and A. L. Soyster, "Modeling and integer programming techniques applied to propositional calculus," *Computers & Operations Research*, vol. 17, no. 6, pp. 561–570, 1990.

[33] K. Kobayashi and K. Hiraishi, "ILP/SMT-based method for design of Boolean networks based on singleton attractors," *IEEE Transactions on Computational Biology and Bioinformatics*, vol. 11, no. 6, pp. 1253–1259, 2014.

[34] Yices SMT Solver: http://yices.csl.sri.com/.

Hindawi
Complexity
Volume 2018, Article ID 9470580, 19 pages
https://doi.org/10.1155/2018/9470580

Research Article

Properties Exploring and Information Mining in Consumer Community Network: A Case of Huawei Pollen Club

Qingchun Meng[1,2] **Zhen Zhang**[2,3] **Xiaole Wan**[2,4] **and Xiaoxia Rong**[2,3]

[1]*School of Management, Shandong University, Jinan 250100, China*
[2]*Research Center for Value Co-Creation Network, Shandong University, Jinan 250100, China*
[3]*School of Mathematics, Shandong University, Jinan 250100, China*
[4]*School of Management, Ocean University of China, Qingdao 266100, China*

Correspondence should be addressed to Xiaoxia Rong; rongxiaoxia@sdu.edu.cn

Received 12 July 2018; Accepted 10 October 2018; Published 6 November 2018

Academic Editor: Yongtang Shi

Substantial changes took place in the role of consumers in the supply chain with the development of practices. They became creators from consumers of product values. More and more consumers express their consumption experiences by posting in network community. Consumer community network is an important place for feedback of product experiences and facilitating product innovation in future. Manufacturers can promote improvement and innovation of products by exploring effective information on the consumer community network, thus improving the experience level of consumers. Therefore, how to explore information in topics (posts) and their relationships becomes very important. Is it possible to describe the structure of consumer community network by complex network and explore information about products and consumers? There is important and positive significance to study the collaborative innovation in the supply chain in which consumers participate. In this paper, the consumer community network was constructed by Boolean retrieve programming and discussed in the methodology and empirical way based on the community data of Huawei P10/P10 Plus. In methodology, interaction difference and uniformity within consumer community were explored by the density of isolated nodes and generalized variance of degree of network. In empirical studies, community network users were divided into ordinary user group, intermediary user group, and enterprise user group according to empirical data, and corresponding interaction networks were constructed. A contrastive analysis on the interaction of these three groups was carried out by combining the existing properties and innovative properties. Topics in each network were put in the order according to significance. Research conclusions have important significance to enrich the network analysis methods, explore the effective information in consumer community network, facilitate product improvement and innovations, and improve the experience level of consumers.

1. Introduction

The social, biological, physical, and technological networks often contain some interactive individuals, which make the complex network, the extension of graph theory, an edged tool to analyze internal structure and dynamic involutions of these networks [1–3]. For example, Boolean network is the combination of the Boolean operation with network structure to solve difficult problems in biological area [4–6]. However, interaction of individuals in the research system of social network services (SNS) [7] has become an important component for rapid high-efficiency propagation of information and discovery of key nodes in the studying networks [8]. The academic circles often abstract the corresponding "nodes" and "edges" from the network data [9, 10] and then construct the network model to analyze its topological properties, including average degree [11], density of graph [12], diameter of graph [13], eigenvector centrality [14], average clustering coefficient [15], etc. This network model not only is conducive to explore deep-layer information like key information propagation [16, 17] and community structure [18, 19] but also helps enterprises in consumer service management [20, 21].

Nevertheless, with the rapid network development, the "Internet +" technologies that combine information technologies arouse the significant impacts of consumers on the market [22, 23]. More and more consumers are active

in expressing their experiences of some bought products or giving purchase suggestions to others [24, 25] in the consumer community formed by different media, including cell phone, PC, or PAD [26]. Manufacturers explore these opinions or suggestions deeply for the purpose of product updating or improvement [27]. Consequently, the role of consumers in the supply chain is changed substantially. They shifted from the consumer of product values into creators [28]. In structure, these consumer communities are more like the derived structures of social network [29]. Therefore, network community has become the important way for communication between enterprises and consumers and information mining [30]. Hence, enterprises shall understand the immediate opinions of consumers in the consumer community, which is very important to develop potential products.

Among consumer electronics, cell phone has become the mobile computer in people's daily life and it is related to various living aspects of users [31]. Moreover, the lifecycle of cell phone is going to be shortened to less than 2 years, which is attributed to the high replacement rate and frequent use [32]. With the progress in informationalization, the brand community and consumer community are developed accordingly. Consumer experience and other information in these consumer communities facilitate the continuous improvement of cell phone in view of some perspective [33]. Currently, Huawei is the leader in the Chinese smartphone market, followed by Xiaomi and OPPO. The market shares of these brands in the fourth quarter in 2017 reached 10.2%, 7.2%, and 6.9%, respectively [34]. They all established their own official consumer communities to exhibit their product design philosophies and accept suggestions from consumers. For example, by May 2018, the number of the published posts on the Huawei Pollen Club about P10/P10 Plus reached 1,556,433 [35], the number of published posts on Huawei P20 reached 261,691 [36], and the number of published posts on Huawei Mate10 reached 1,293,712 [37]. These posts covered tremendous product information and experiences of consumers [38].

Considering the extreme importance of consumers to manufacturers, what is the structure of consumers in the web community? What cell phone topics are different users concerned with in the community? Which connections are there between different topics and how strong such connection is? What characteristics are there in the changes of topics as time goes on?

On this basis, this paper is going to explore data from the Huawei P10/P10 Plus community to address above problems mainly from methodologies and empirical studies. In methodology, interaction difference and uniformity among different consumer community networks as well as key time points of the network dynamics were explored by the density of isolated nodes, generalized variance of degree of network and node sequential emergence determination coefficient. In empirical studies, community network users were divided into Ordinary User Group (OUG), Intermediary User Group (IUG), and Enterprise User Group (EUG) according to empirical data, and corresponding interaction networks were constructed. A contrastive analysis on the interaction of these three groups was carried out by combining the existing properties and innovative properties. Topics in each network were put in the order according to significance. Secondly, the emergence law of cell phone topic lifecycle was analyzed by combining the theory of product lifecycle with node sequential emergence determination coefficient.

The remainder of this paper is organized as follows. Section 2 is literature review on existing research methods. Section 3 extracts topics and classifies users according to posting data of Huawei P10/P10 Plus users. Section 4 constructs the complex network models for three types of users, respectively. Some new properties, such as density of isolated nodes, generalized variance of degree of network, and node sequential emergence determination coefficient, are proposed. A statistical analysis is carried out by combining these new properties with the traditional statistical properties. Section 5 further analyzes "leaders" in networks and explores information like closeness and significance of topics. Section 6 elaborates enlightenments to enterprise management which are gained from empirical analysis. Finally, the corresponding sketch of methods is displayed in Figure 1.

2. Literature Review

There are rather more literature reports about the consumer interaction from 4 ways to explore the law behind it, which are stated below.

2.1. Consumer Interaction. Many studies on consumer interaction have been reported worldwide. Georgi and Mink (2013) explored the positive impacts of interaction of electronic (online) consumers on performance of innovative enterprises [39]. Smaliukiene et al. (2015) analyzed consumers' discussions of network construction in the online forum provided by suppliers when they studied the online tourism service, finding that consumer interaction was conducive to analyze procedures in global online tourism service departments effectively [40]. Bruhn M et al. (2014) performed the online investigation of three virtual B2B brand communities and verified the positive effect of consumer interaction on brand loyalty by an empirical study [41]. Millán et al. (2016) analyzed the impacts of consumer interaction on satisfaction to vocation by the fuzzy qualitative method, finding that quality, strength, value, and influence of consumer interaction are important conditions of vocation experience [42]. Based on 821 samples, Wei et al. (2017) discussed the fundamental mechanism of influences of consumer interaction on experiences of participants. They reported that specialized knowledge communication and social emotional support during the consumer interaction are vital to the implementation of activities of service providers [43]. Chen et al. (2011) discussed the influence of customer interaction on the relationship quality between service companies and customers by constructing a conceptual model and found that such relationship quality can be improved by improving the consumer interaction methods [44]. However, it is easy to know that all above studies are mainly macroscopic analyses on consumers' behavior based on survey questionnaire but neglect the difference among different consumer community

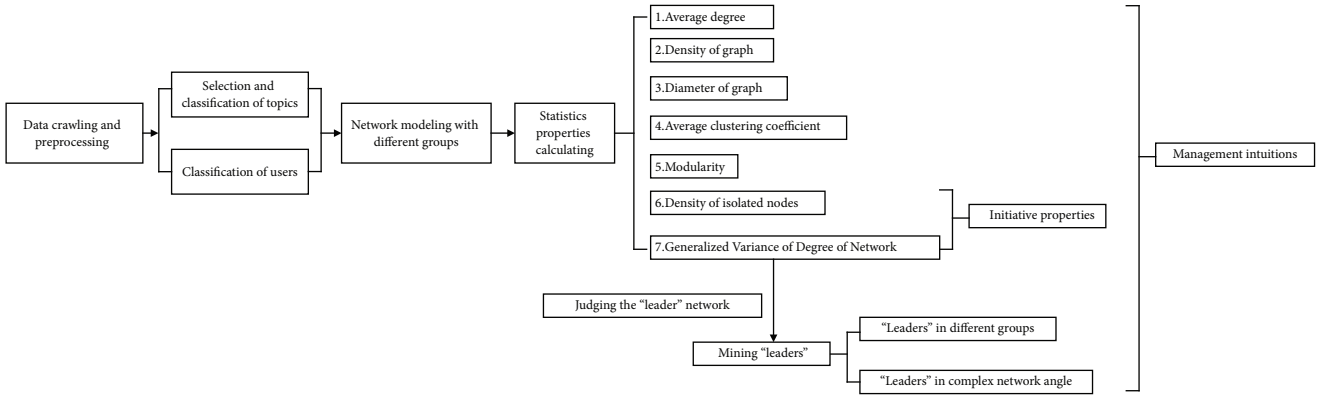


Figure 1: Sketch of the exploration study.

networks. Few scholars have discussed differences of interaction contents in the community brought by changes of product lifecycles.

2.2. Research Methods of Consumer Online Community. From the view of methodology, there are mainly four methods in research of customer interactive behavior containing statistical methods, structure equation modelling, experiment and case study and complex network analysis, which are stated as below.

In method of statics, Oh., et al. (2015), classified the test subjects of 315 university students as three groups and conducted two-way ANOVA to test the hypotheses of the research model [45]. Zollet and Back (2015) collected data from 138 firms in Switzerland and Germany and analyzed with multiple regression analysis [46]. Khan et al. (2016) analyzed 1,922 brand posts from five different brands of a single product category in three different countries and used ordinary least square and hierarchical moderation regression to test the hypotheses [47]. Nourikhah and Akbari (2016) used Bayesian data analysis with a generalized linear model (GLM) to estimate the overall satisfaction of the users in the form of the posterior distribution of opinions [48]. Wan et al. (2016) introduced least squares support vector machine (LS-SVM) innovatively into the study on consumer electronics supply chains [23]. These studies took consumers as a whole, then from the perspective of the supply chain or enterprises, analyzing consumers' interaction impact on supply chain or their features. However, consumer network is not a simple whole, but a complex structure, which meets the structure of the general complex network and has its own characteristics at the same time.

Many scholars introduce structure equation modelling method to study consumer behavior in online community. Shobeiri et al. (2014) used structural equation modelling based on EQS 6.1 to assess the measurement and structural models [49]. Liou et al. (2015) adopted structural equation modeling to investigate the factors that influence users' use intentions regarding broadband television [50]. Islam and Rahman (2017) analyzed the data using structure equation modelling through a questionnaire survey of 430 Facebook users [51]. The structure equation modelling can explain features in customers interactive network; however it also ignores the structure of the customers communities which would leave out some detailed information like the important topics and customers.

In terms of experiment and case study method, Kilgour et al. (2015) employed depth interviews initially, followed by questionnaires, and then computer assisted content analysis was performed on 723 online media articles relating to social media marketing to identify semantic and conceptual relationships [52]. McKechnie and Nath (2016) explored this issue in an online experiment with 273 subjects browsing 4 websites offering identical products but with variable levels of interactivity and personalization features [53]. Chu et al. (2017) conducted two experiments to identify an effective communication strategy to facilitate social media marketing using a combination of communication facets such as frequency, direction, formality, and content [54]. Firstly, in these papers, experiment and case studies were conducted within a confined condition, which means that the participants are easy to be interrupted by some other reasons. Secondly, participants and case study could not represent the whole interactive network to some extent.

In complex network analysis method, Chiang and Wang (2015) extended research on the interactive features of product-review networks by considering the out degree centralization, density, and microstructure of product-review networks [55]. Li and Gu (2015) proposed an OSN link formation model from the perspective of user behavior, which reproduced degree distribution, clustering and degree correlation of OSN [56]. Andersen and Mørch (2016) classified user types through social network statistical analysis and constructed "user-topic" hybrid network with user interaction analysis of user posts [57]. Baumgartner and Peiper (2017) extended a novel method called stochastic block modeling to derive communities of cannabis consumers as part of a complex social network on Twitter [58]. Liu et al. (2017) proposed a complex network model with reviews as nodes by calculating reviews topics with latent Dirichlet allocation model and topic similarities among reviews with Pearson similarity [59]. These studies consider it from a complex network view, ignoring the statistics characteristics between the same type networks in different periods.

Table 1: Classification of topics.

Type	Topics
System	System, Upgrade, Lock screen, Unlock, Black screen, Font, Resolution, Update, Lightness, Color, Screen capture, Location, Telephone, Net, Mode, Vague, Data, WIFI, Power off, Beta, Theme, Ring, Voice assistant, Heat, Wall paper, Desktop, NFC, Root, 4G, Internet speed, GPS, Position
Software	WeChat, Fingerprint, Transposition card, Consumption, Message, Game, Flash back, Program, Code, Backups, Music, Defrayment, Video, King Glory, Jingdong, Vmall, Weibo, QQ
Hardware	Life, Taking pictures, Power consumption, Charging, Memory, Pick-up hand, Camera, Light, Anti-fingerprint oleophobic coating, Battery, Home key

In our paper, data were scrawled from club.huawei.com, which enable us to avoid interview effects [60] and some other possible negative influence accompanying survey research [61, 62]. Later we will clean the data and build complex model.

3. Data Crawling and Preprocessing

3.1. Data Source. For topic type, Huawei community, Xiaomi community, and OPPO community emphasize on different topics. For example, the OPPO community focuses on camera performance of the phone. Huawei P10/P10 Plus community has relatively more topics, covering hardware, software, system, appearance design, and even price. More importantly, the community has stronger data integrity and accessibility. Although the Xiaomi community has many topics, it only displays the latest data, which were not as comprehensive as that of Huawei P10/P10 Plus community since February 2017. Hence, post data in the Huawei P10/P10 Plus community were collected in this paper for information mining by complex network.

In this paper, post data in the Huawei Pollen Club (HPC), a consumer community formed by Huawei P10/P10 Plus, from February 8th, 2017, to November 4th, 2017, were collected [35]. Members of the club participated in communication of relevant products after registration. In this club, consumers can raise questions and interact with others by replies, thus increasing understanding on Huawei products. On the other hand, Huawei can make responses in time, help them to solve ticklish questions, and explore problems that consumers are highly concerned according to consumers' information feedback in the community, thus enabling improvement of products during upgrading and increasing profits of the enterprise.

3.2. Initial Data Screening. A total of 125,163 data were collected initially, covering titles and contents of posts (excluding replies), user name, user level, and publishing time. Since user browsing or reply was updating dynamically and generated continuously during data acquisition, it was inevitable to generate some repeated data. In this paper, the latest state of the same post was applied. After selected operation, 78,320 data were retained. Later, 824 invalid data of banned to post, banned to login and shielded data because of advertisement and unrelated information were further eliminated. Finally, 77,496 valid data were kept.

3.3. Data Analysis and Processing. In this section, data were analyzed from extraction of hot topics and user classification, which prepares for the construction of weighted network model in Section 4.

3.3.1. Extraction of Topics. Firstly, the core topics are extracted from what the users consider. Most of the data presented by users on the website are in the form of posts. It is necessary to extract the topics from the post in order to learn the needs of the users.

Through calculating the frequency of the topic combined with the features of phone via programming with Boolean operation to judge whether the topics occur in the post or not, 100 topics are selected (see Appendix A). After sorting the higher frequency ones, they are divided into three parts including system, software, and hardware, according to their feature, showed in Table 1.

3.3.2. Classification of Users. In order to specify interaction and different topic focus within community, users of the HPC can be divided into three groups according to functions and roles [63–68], namely, OUG, IUG, and EUG. The OUG refers to users who bought Huawei products and registered in the HPC. The IUG refers to users who have received official training of Huawei and are willing to answer questions of other users. The EUG refers to the official enterprise employees, covering technicians, salesmen, and publicists. Level labels and meaning of each group are listed in Table 2.

A statistical calculation on posting frequencies of all users of each level showed in Table 2 was made (see Appendix B), getting proportions of posts of three user groups in Figure 2.

It can be seen from Figure 2 that 99% posts were published by ordinary users, indicating that OUG is the main force. However, it still cannot replace the key role of the rest two groups in the community. Hence, different models were constructed to the OUG, IUG, and EUG, respectively.

4. Weighted Network Analysis

In this part, this paper introduces complex network analysis method. The nodes denote 61 topics in Table 1, and if a user mentions two topics K_i and K_j, in a post title and text at the same time, it suggests that there is a close relationship between these two topics, which corresponds with an edge between nodes i and j. This step is achieved by Boolean retrieve in programming. The weight of edges denotes the number of users. That forms undirected weighted network.

Table 2: Meaning of HPC.

Group name	Level name	Meaning of levels
OUG	Newcome	OUG level 1
	Beginners	OUG level 2
	Preliminary learners	OUG level 3
	Small success	OUG level 4
	Further progress	OUG level 5
	Master	OUG level 6
	The dedicated	OUG level 7
	The self-contained	OUG level 8
	Great success	OUG level 9
	Pinnacle	OUG level 10
	Magic master	OUG level 11
	The matchless	OUG level 12
	Limited member	Limited use due to long unregister or other reasons
IUG	Hot fans	Activating area atmosphere and eager to answer the questions of other users
	Expert fans	Willing to experience the latest products and ROM, positive feedback problems during use with good language organization, having enough time to participate in product evaluation, and enjoying taking pictures and reading experience
	Female fans	Special female members dedicated to women's topics
	Internal manager	On the basis of all Pollen member, an independent special user group with management authority
	Internal expert	Application for internal test, an independent special user group, with members of internal test core group
	Pollen director of city	The core link of regional Pollen fans and participating in Huawei's deep marketing decision in the region
	Pollen director of universities	The university club management of Huawei, assisting Huawei in the publicity and personnel recruitment
	Special forces of HPC	A group of technical master trained by the Huawei for researching phone, sharing information and solving problems for others
	Moderator of HPC	Management in various articles of the forum, and promoting the healthy development of the forum
	Moderator of Huawei Pollen Sub-club	Management in a group of forums
	Moderator of game center	Management in game forums
	HRT team	Providing experience of third party Rom version based on official Rom or other vendors
	Super-circle director of HPC	Maintaining circle order, activating circle discussion and discussion atmosphere, and establishing good communication environment for Pollen members
	Theme fans	In order to get all the pollen to have a better experience, modifying the theme, making a custom theme and so on
EUG	Pollen group	Focusing on the Huawei mobile phone evaluation and guidance, solving common problems, enhancing the interaction of pollen
	Pollen group	Focusing on the Huawei mobile phone evaluation and guidance, solving common problems, enhancing the interaction of pollen
	HPC team	The official management team of the club
	Official team	Huawei official team
	EMIUI product manager	Official product manager for the EMIUI system
	EMIUI official team	The official team for developing the EMIUI system
	Product manager	Huawei product manager

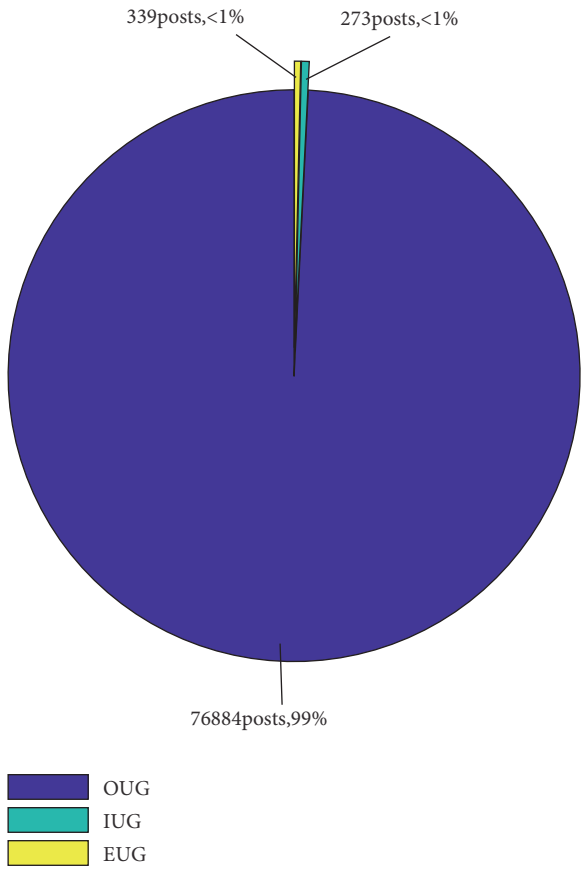


Figure 2: The ratio of posts of different groups.

Because different groups of users have different positions in the network, they play different roles. Therefore, this paper establishes networks according to users' groups.

4.1. Modeling of Networks. By using Gephi software, we get three groups of users' interaction networks, respectively, in Figures 3, 4, and 5. In the graphs, the nodes of the same color represent the same kind of community [69, 70]. The size of the nodes represents the eigenvector centrality, that is, the power to control other topics. The color of the edge represents the number of people who focus on two topics at the same time. The deeper the color (purple) is, the more the people who are concerned about the two topics are, which shows that these two topics have strong correlation.

The interaction network of OUG is shown in Figure 3 Topics are divided into five communities: "Taking pictures," "System update and battery," "Fingerprint unlock," "APP," and "Internet speed," which reflects the system problems, software problems, and hardware problems users are concerning. However, the connection between "System" and "Update" has the deepest color among all topics' edges, indicating the high frequency of simultaneous mention of these two topics by users. This implies that the cell phone problems might be brought by system updating. In addition, "WeChat" is closely related to the "APP" community and topics of other communities, indicating that "WeChat" is the core application of OUG.

It is easy to note that edges in the network have relatively uniform color, which implies that users concern extensive problems. Besides, the OUG often proposes their questions by posting in the community and make partial or complete effective answers to problems of other users. They have strong uncertainty.

The interaction network of IUG is shown in Figure 4. The network is divided into two communities: "System applications" and "Hardware." Although problems still involve system, software, and hardware of cell phone, the system and applications are divided into one community, indicating that the IUG can classify topics effectively. Compared with the OUG, the IUG is aware of problems that the OUG has not noticed. For instance, "Pattern" is just a periphery topic in the interaction network of OUG, but it is a core topic in the interaction network of IUG and highly related to other topics.

Compared with the interaction network of OUG, edge color in the interaction network of IUG is not uniform. Many edges have deep color, especially in the "System applications" community. The IUG associates key topics that users are discussing effectively according to users' questions and offer corresponding answers. They fulfill the responsibility of answering questions authorized by the Huawei community.

In Figure 5, the interaction network of EUG is also divided into 3 communities: "System updating," "Taking pictures," and "Software applications." In the "Taking pictures" community, edges between any two topics have relatively deeper color, indicating that the Huawei officials pay attentions to propagation of the camera performance of cell phones. This is because Huawei officials regularly encourage OUG to exhibit their own pictures. Moreover, the topic "System" is strongly correlated with other topics.

Obviously, the IUG answers questions of users and summarizes topics. Based on the IUG, the Huawei officials answer questions related to "System," "Upgrade," and "Update." They also answered the "WeChat" problems that users are concerned. In other words, the EUG can not only guide the discussion themes in the community by observing the OUG and IUG but also answer problems of the OUG accurately.

By comparing these three networks, three characteristics are recognized:

(1) The number of hotspots of core topics increases gradually. The node size in networks represents the significance. Node size in the interaction network of OUG is more uniform than that in the interaction network of IUG, indicating that the OUG has more questions in both quantity and complexity. However, the IUG and EUG with experiences can explain topics specifically, thus increasing the number of core topics relatively. The concerned problems also present targeted variation.

(2) There are significant differences among different communities. The difference among different user groups is manifested by the number and members of communities. Just as definitions of IUG, it is mainly to classify problems of the OUG and give specific answers. Therefore, it only involves two communities. The EUG will cooperate with concerned points of the OUG and make corresponding

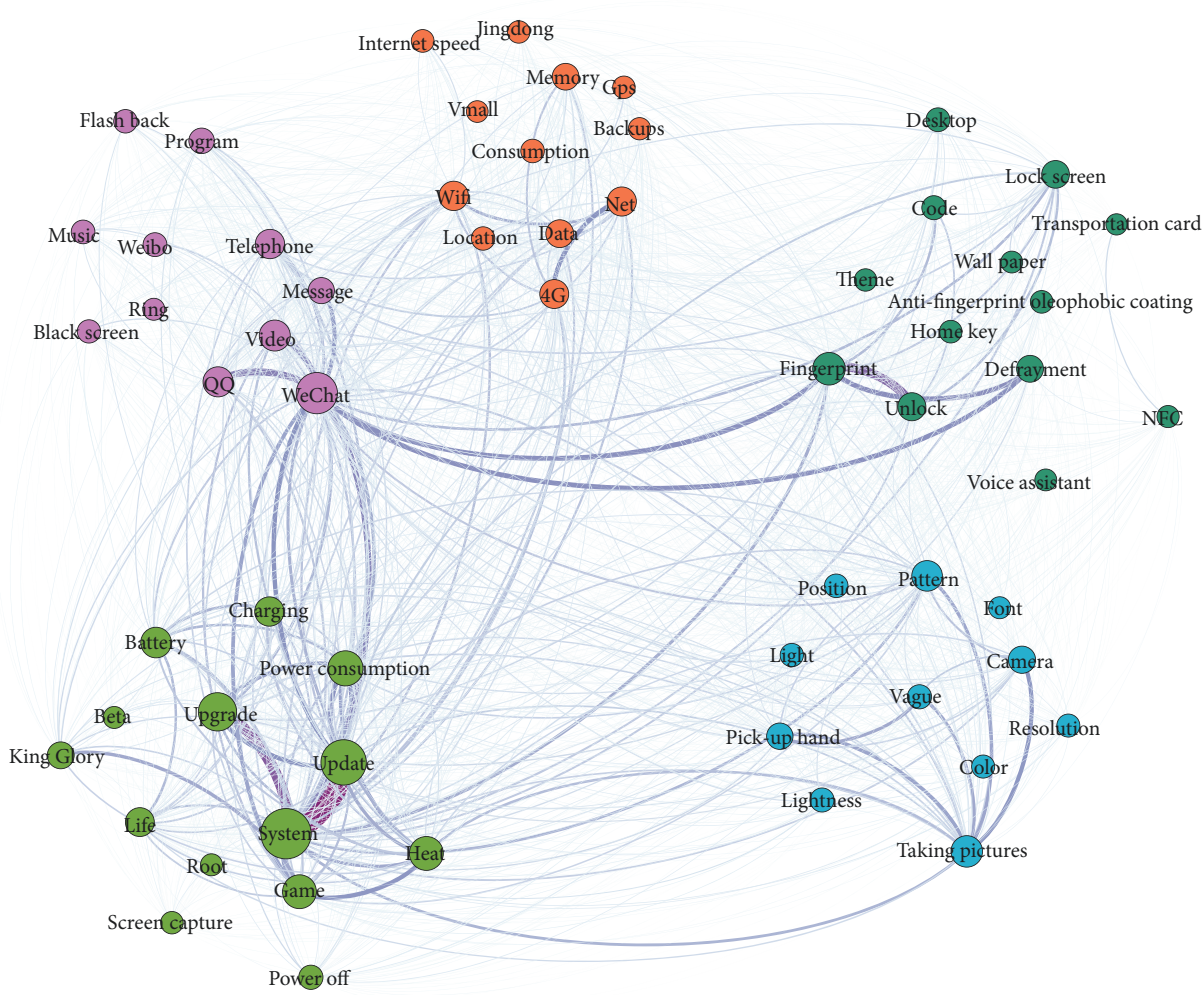


Figure 3: OUG interaction network.

guidance. Therefore, these two groups have similar number of communities. However, these two groups have certain differences in communities' members, which is caused by their different cognition degree on correlation degree of problems.

(3) The correlations of topics are significantly different. The OUG concerns all aspects of cell phone, because they have poor knowledge on roots of cell phone problems. Therefore, edges have relatively uniform color. In contrast, the IUG understands relevant problems of cell phone well. It highlights connections of different types of problems during reasonable standardizing of problems. The EUG is mainly to answer most questions of the OUG and propagate the system and unique camera performance of cell phone. Hence, only edges within these two communities are relatively deep.

4.2. Statistical Analysis of Networks. From the former descriptive analysis of three networks, the difference between them will be quantified by using complex network properties: let G(V,E,W) be a nonempty weighted graph with $|V| = n$ and$|E| = m.A = (a_{ij})_{n\times n}$ is the adjacency matrix of G, in which a_{ij} is 1 if node i and node j are connected and 0 otherwise. Similarly, $A^w = (a^w_{ij})_{n\times n}$, is the weighted adjacency matrix of G, in which a^w_{ij} denotes the weight of the edge between node i and node j. W_{all} represents the sum of the weight of the edges in G. Through comparing the statistical properties between the constructed networks and relative null models, which includes average degree, density of graph, average clustering coefficient, diameter of graph, modularity, and initiative ones, containing density of isolated nodes, generalized variance of degree of network, we can specify the information value of networks, in which the null model denotes $G(V', E', W')$ with $|V'| = |V|$ and $W_{all} = W'_{all}.W'_{all}$ is the he sum of the weight of the edges of null model.

This paper use following statistical properties [69]: (1) Average degree: average degree, denoted $\langle k\rangle$, describes the mean of all nodes in the network. In this paper it represents the average of topics' relative topics. (2) Density of graph: density of graph, ρ, is the ratio of the existing number of the edges m to its maximum possible number of edges. We use it to detect the density of topics network. (3) Diameter of graph: the diameter of graph, denoted by d_{max}, is defined

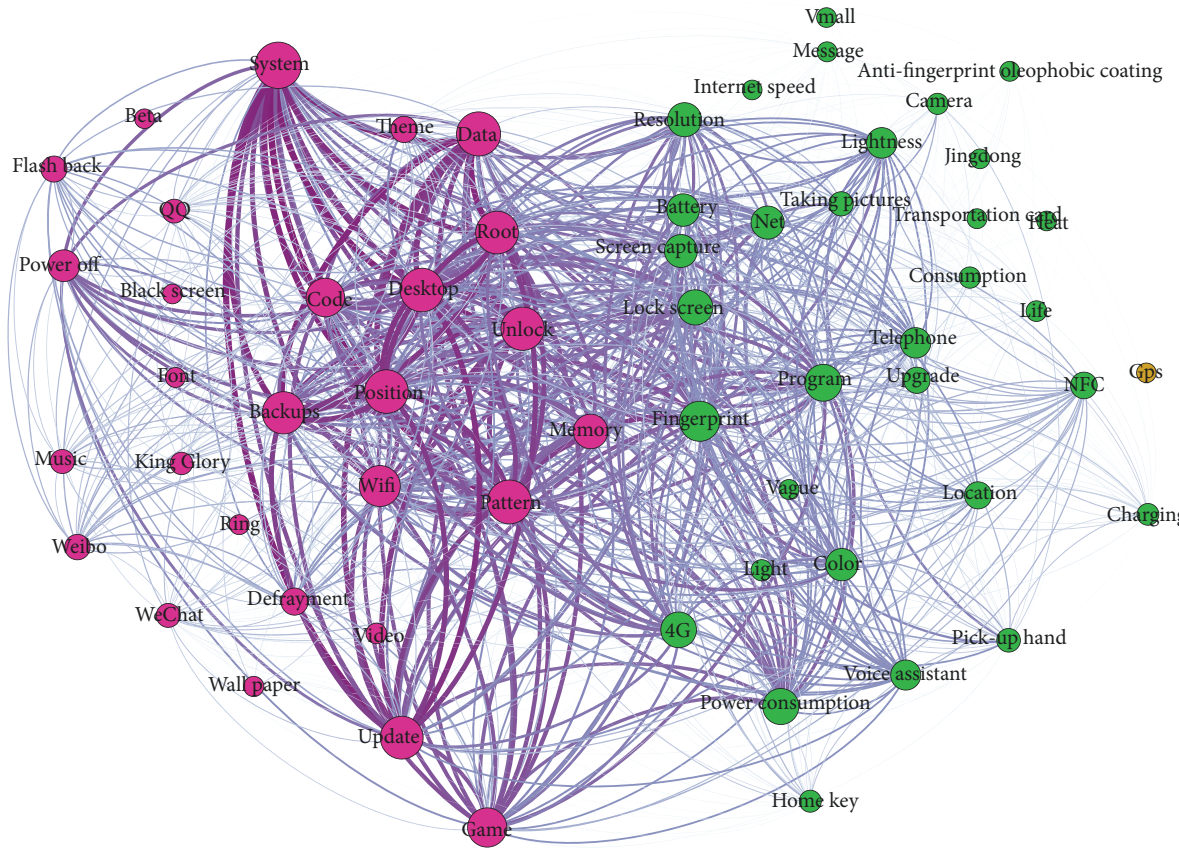


Figure 4: IUG interaction network.

to the largest of all distance in the graph. The smaller d_{max} is, the more stable network would be. This property can describe the closeness of the topics. (4) Average clustering coefficient: if there are edges between each two of nodes i, j, and k, then it forms a triangle. Thus, average clustering coefficient C is defined as the ratio of such triangles in graph G. It can describe the local stability of network. (5) Modularity is a measure of the level or degree to which a network's communities may be separated and recombined, which is a commonly used criterion for determining the quality of network partitions. It can classify the topics according to their associations. Although they can describe the general features of different networks, it is still necessary to measure the following features of networks.

The quantitative description of isolated topics and judgment rule of "leaders" network in weighted networks, which has nodes with special important status called "leaders" in it. The OUG network has no isolated topics; however, different circumstance occurs in the IUG and EUG, which means topics have different status among three groups. What is more, there are some researches on the judgment rule of "leader" network in unweighted networks but no one in weighted network like topics networks in this paper. Mining the "leaders" can guide EUG focus on important topics. If company can solve the timely, other problems would be modified.

Due to these demand for research, this paper proposes the following properties to dig out the properties of the networks.

4.2.1. Density of Isolated Nodes. Namely, isolated node i satisfies

$$\sum_{j=1}^{n} a_{ij} = 0 \quad (i \in \{1, 2, \ldots, n\}) \tag{1}$$

Therefore, density of isolated nodes p_a is defined as the ratio of the number of isolated nodes n_a to n. That is,

$$p_a = \frac{n_a}{n} \tag{2}$$

p_a measures the connectivity of network, which means that the more connective network is, the smaller p_a it would be. Traditional network analysis based on good connectivity of graph, lacks of this property. It can explore the difference due to interaction level between user groups; as a result some topics become isolated ones.

4.2.2. Generalized Variance of Degree of Network. Considering the nodes in undirected network, the generalized degree of each node is the sum of the weight of links between their neighbors. So the variance of all generalized degree in graph is the generalized variance of degree of network. It can judge whether G has "leaders" compared to the null model. We can use it to detect whether the network has "leaders" or not, as well as the uniformity of weight distribution. The thought of its definition comes from variance of degree of undirected unweighted graph [71]: if the degree of node i denotes d_i,

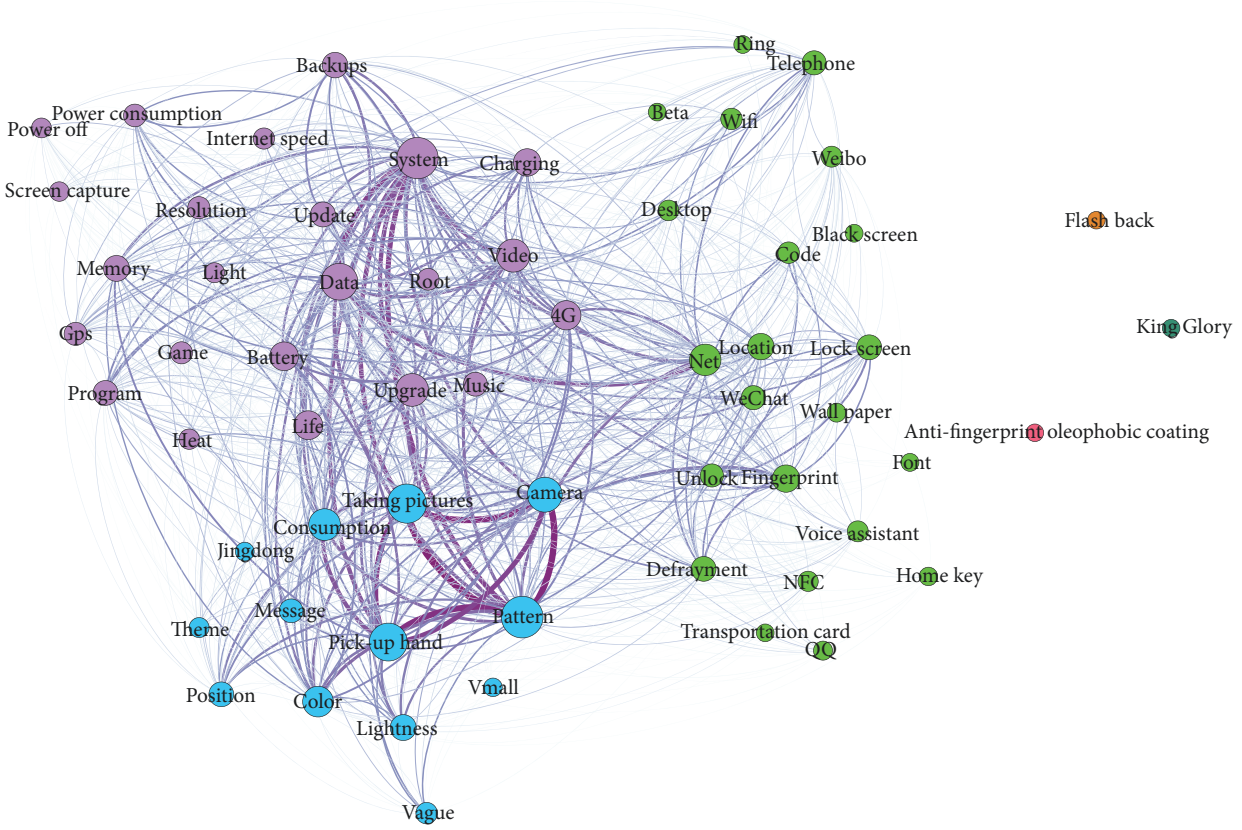


Figure 5: EUG interaction network.

the expectation of average degree is $E_x(G) = \sum_{i=1}^{n} d_i/n$, so variance of degree is

$$\mathrm{Var}(G) = \frac{1}{n}\sum_{i=1}^{n}\left(d_i - E_x(G)\right)^2 \tag{3}$$

However, if G is an undirected weighted graph, variance of degree ignores the weight of edges' impact on the uniformity of G. As a result, this paper defines generalized variance of degree: if n_i neighbors of node i denotes d_i^g,

$$d_i^g = \begin{cases} \sum_{l=1}^{n_i} w_i^l & n_i > 0 \\ 0 & n_i = 0 \end{cases} \tag{4}$$

where w_i^l is the weight of link between node i and its l^{th} neighbor node, similarly the expectation of average of generalized degree is $GE_x(G) = \sum_{i=1}^{n} d_i^g/n$, and generalized variance of degree in network can be computed as follows:

$$G\mathrm{Var}(G) = \frac{1}{n}\sum_{i=1}^{n}\left(d_i^g - GE_x(G)\right)^2 \tag{5}$$

This property is the general form of variance of degree, if G is undirected unweighted graph generalized variance of degree degenerates as variance of degree, through calculating (3) and (5).

Moreover generalized variance of degree of its relative null model $G(V', E', W')$ is $G\mathrm{Var}(G)^{zero}$ and the standard deviation of $G(V', E', W')$ is $\sigma[G\mathrm{Var}(G)^{zero}]$. Since generalized variance of degree of $G(V', E', W')$ matches the Z distribution, in Appendix C Z distribution is proved to approximate normal distribution. In this paper, "leader" network is defined as follows: if the generalized variance of degree of G is bigger than the "$3-\sigma$" margin of null model $G(V', E', W')$; that is, $G\mathrm{Var}(G) > \mathrm{Mar}(G) = E[G\mathrm{Var}(G)^{zero}] + 3\sigma[G\mathrm{Var}(G)^{zero}]$, or "autonomy" network otherwise. From this definition, "leader" network has significant nodes named "leaders" controlling other nodes and influencing the generalized variance of degree, however the importance of "autonomy" network is relatively even.

$G\mathrm{Var}(G)^{zero}$ may change every time along with the different result of random construction. If the generalized variance of degree of p^{th} random result is $G\mathrm{Var}(G)^{zero}{}_p$, and there are N random graphs; then below forms

$$\begin{aligned} E\left[\frac{1}{N}\sum_{p=1}^{N} G\mathrm{Var}(G)^{zero}{}_p\right] &= \frac{1}{N}\sum_{p=1}^{N} E\left[G\mathrm{Var}(G)^{zero}{}_p\right] \\ &= \frac{1}{N}\sum_{p=1}^{N} E\left[G\mathrm{Var}(G)^{zero}\right] = E\left[G\mathrm{Var}(G)^{zero}\right] \end{aligned} \tag{6}$$

From this equation, it shows that a large number of generalized variance of random networks can be computed

Table 3: Statistical results of three networks.

Group name	Average degree	Density of graph	Diameter of graph	Average clustering coefficient	Modularity	Density of isolated nodes	Generalized variance of degree of network
OUG	57.705	0.962	2	0.971	0.150	0.00%	7620144.613
IUG	33.902	0.565	3	0.840	0.083	1.64%	265766.736
EUG	32.492	0.542	3	0.827	0.140	4.92%	15014.359

Table 4: The results of null models.

Group name	Average degree	Density of graph	Diameter of graph	Average clustering coefficient	Density of isolated nodes	Generalized variance of degree of network
OUG	60	1	1	1	0.00%	2977.656
IUG	60	1	1	1	0.00%	566.147
EUG	60	1	1	1	0.00%	136.826

to get the expectation of it. When N is relatively big, the mean of generalized variance of random networks of N random graphs can approximate $E[G\mathrm{Var}(G)^{zero}]$.

4.3. Comparison Analysis. The numerical result of 5 traditional properties showed in Section 4.2 (1)-(5) and 2 initiative ones are displayed in Table 3:

From OUG to IUG and EUG, in terms of the average degree, the relationships among topics decrease gradually. The density of them changes from great density to sparseness. Especially the proportion of from isolated topics and Figures 3–5, there is no isolated topic in OUG but an isolated topic "GPS" in IUG and three in EUG including "King Glory," "Anti-fingerprint oleophobic coating," and "Flash back." This shows that the IUG solves three problems of ordinary users through interaction, so three topics become isolated in the EUG network. However, IUG does not solve "GPS," so EUG makes relevant interpretation. In terms of diameter of graph, the OUG is more compact compared with other two networks, suggesting that users in OUG equally focus on topics not having a clear mind on their relation. The average clustering coefficient illustrates microstructure of three networks, because the IUG and EUG have content knowledge, reducing the number of unnecessary contact between topics. Modularity shows the rationality of the division of the communities of three networks. Generalized variance of degree of network indicates that all the three groups have some provocation opinions on the relations between the topics, some topics holding more attention compared with others. These topics with significant status are mined in Section 5.

5. Information Mining of "Leaders"

Fu et al. (2016) suggested that nodes, which hold great importance, having strong relationship with others in the network, are called "leaders" [72]. This paper also judges whether there are "leaders" in three networks by computing properties results with their corresponding null models respectively. Moreover, "leaders" and closeness of topics are analyzed via eigenvector centrality method.

5.1. Existence of "Leaders". Firstly, whether networks have "leaders" that are judged: 1000 random networks are established by Matlab programming according to each null model structure separately. The 6 properties of the 1000 random networks are as follows.

As it shows in Table 4, since the characteristics of the networks built in this article are: $W_{all} > n(n-1)/2$, which results in the mean average degree of all random networks corresponding to three groups are 60. By comparing the results between Tables 3 and 4, it is found that their properties have a significant difference.

If the generalized variance of degree of OUG network, IUG network, and EUG network is denoted, respectively, by $G\mathrm{Var}_{OUG}, G\mathrm{Var}_{IUG}, G\mathrm{Var}_{EUG}$ and the margin value of their corresponding null models is denoted by $\mathrm{Mar}_{OUG}, \mathrm{Mar}_{IUG}, \mathrm{Mar}_{EUG}$, the following results are obtained from computing: $G\mathrm{Var}_{OUG} = 7620144.613 \gg \mathrm{Mar}_{OUG} = 4652.730, G\mathrm{Var}_{IUG} = 265766.736 \gg \mathrm{Mar}_{IUG} = 857.300, G\mathrm{Var}_{EUG} = 15014.359 \gg \mathrm{Mar}_{EUG} = 211.174$. It shows that the generalized variance of degree of three networks is greater than "$3-\sigma$" boundary of that of their null models. So the OUG network, IUG network, and EUG network are "leader" networks with significant "leaders." The "leaders" in these networks will be explored below.

5.2. Finding "Leaders" in Certain Network. After certifying networks with "leaders," this section will dig them out and analyze the closeness of topics by eigenvector knowledge.

Iranzo (2016) analyzed the financial ability of village [73], so the importance of topics is also calculated by this method in this section. The concept of eigenvector centrality is that the importance of every node in network is associated with the number and quality (importance) of its neighbor nodes.

The results of eigenvector of maximum eigenvalue of three networks are calculated and normalized by Matlab, denoting $x_i^c = x_i / \sum_{j=1}^{n} x_j$, shown in Appendix D. Fu et al. (2016) proposed that the top 10 percent of importance of all nodes are "leaders" [72] and there are 61 topics in this research, so the "leaders" of three networks are in Table 5.

From Appendix D and Table 5, it is obvious that there is a certain difference among the importance ranking of

Table 5: "Leaders" of three networks.

Group name	Topics
OUG	System, Pick-up hand, WeChat, Upgrade, Power consumption, Voice assistant
IUG	System, Data, Memory, Wallpaper, Location, Unlock
EUG	Memory, System, Taking pictures, Resolution, Data, Update

Table 6: Kendall coefficient test.

Comparison groups	OUG vs IUG	IUG vs EUG	OUG vs EUG	Overall
Kendall coefficient	0.694	0.723	0.797	0.651
P value	0.025	0.013	0.002	0.001

three networks topics and "leaders", which means that the interaction levels of users on website are different and cause the difference in core topics. Kendall coefficient test in nonparametric statistics are computed in Table 6 to see whether the rankings of topics importance of three groups are different or not.

From the results of Kendall coefficient, the overall consistency is relatively low. Moreover, in the comparisons between two among them, the consistency of ranking "OUG vs IUG" is lowest compared with other two pairs. And the highest consistency pair is "OUG vs EUG." From the results of P value in Table 6, the Kendall coefficient is reliable under 5% significant level.

As is analyzed in the former part, EUG need to combine the feedback of IUG, for example, system version test and solution to the problems, with taste of OUG to issue content, so it has a relatively high consistence with OUG in ranking. However, the classification of topics of IUG makes it different from others.

In computing eigenvectors centrality progress, we can also get the maximum eigenvalue λ_1 of corresponding networks, which means the interaction intensity of the network. If the maximum eigenvalue of the OUG network, IUG network, and EUG, respectively, is denoted by $\lambda_{OUG}, \lambda_{IUG}, \lambda_{EUG}$, the values of them are calculated: $\lambda_{OUG} = 5651.53 \gg \lambda_{IUG} = 1045.26 > \lambda_{EUG} = 267.81$. From the maximum eigenvalues of the three networks, it is shown that the maximum eigenvalue gradually declines from the OUG to the IUG and then to the EUG. Because the OUG has 99% valid data, the maximum eigenvalue of the network is naturally large. However, the interaction effect of IUG network is better than that of the EUG network. This also proves that intermediary users have reduced the pressure of the EUG as the backbone.

5.3. Ranking Topics Based on Multiplex Network. Importance of topics in all groups is analyzed. However, three networks have overlapping topics. To get all topics that users concerned, that is, to discover the overall "leaders," three networks are overlapped effectively in this section and analyzed from the perspective of multiplex.

Firstly, the multiplex network is designed as follows. The bottom layer is the interaction network of OUG, the middle layer is the interaction network of IUG, and the top layer is the interaction network of EUG. Same topics in two adjacent layers are connected, getting Figure 6.

In Figure 6, the pink nodes represent isolated topics in the corresponding layer and the grey nodes are interactive topics. Larger nodes (topic name) reflect higher degree. Clearly, the IUG fails to solve "GPS" and thereby generates the isolated topics. However, these topics are absent in the isolated topics of interaction network of EUG, indicating that EUG solves problems beyond the competence of OUG and IUG. Similarly, three isolated topics in the interaction network of EUG are absent in the interaction network of IUG, which reflects that these three topics are solved by the IUG and the EUG does not need to explain them. In brief, the IUG, the bridge between the OUG and the EUG serves as the "problem filter" well. They enhance the ability of users to solve problems through interaction and relieve pressure of customer service of enterprises. This also reflects that stimulating the interaction between the IUG and the OUG can bring consumers fast updating in product experience.

Boccaletti et al. (2014) introduced the method of ranking of node importance based on multiplex network. If normalized eigenvector centrality of OUG network, IUG network, and EUG network is $(x_1^{c(i)}, x_2^{c(i)}, \ldots, x_n^{c(i)})^\tau$ $(i = 1, 2, 3)$, then the j^{th} $(j = 1, 2, \ldots, n)$ node importance is defined below [74]:

$$x_j^c = \sum_{i=1}^{3} x_j^{c(i)} \tag{7}$$

By combining results in Appendix D and (7), the importance and its ranking of topics based on multiplex network are shown in Appendix E. "Leaders" in the multiplex network are "system," "memory," "camera," "data," "update," and "pixel." However, "Anti-fingerprint oleophobic coating," "King Glory," and "Fingerprint" are less correlated with other topics, indicating their less importance. These topics have certain difference in importance.

6. Conclusions

The consumer community network is explored in this paper by methodology and empirical study based on the data in Huawei P10/P10 Plus community. In methodology, interaction difference and uniformity within consumer community are explored by the density of isolated nodes and generalized variance of degree of network. In empirical studies, community network users are divided into OUG, IUG, and EUG

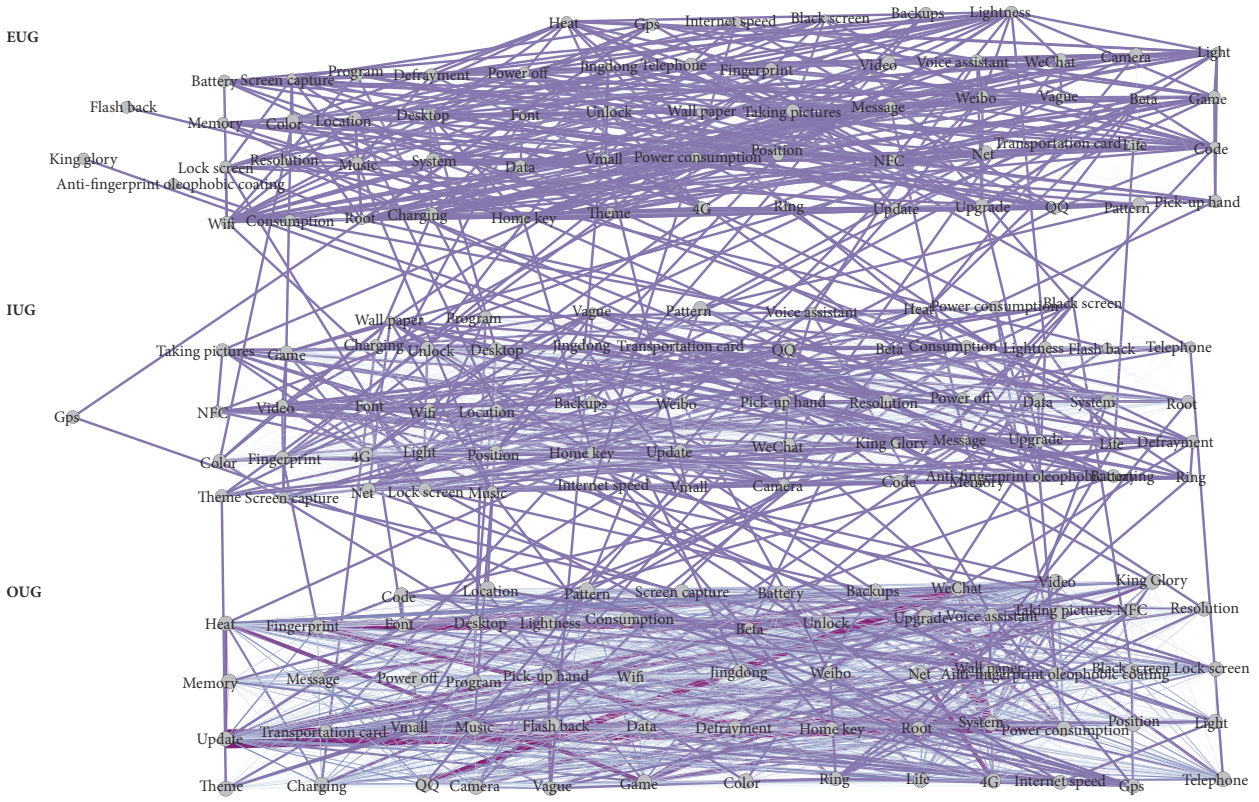


Figure 6: Consumer network based on multiplex network.

according to empirical data and corresponding interaction networks are constructed. A contrastive analysis on these three interaction networks is carried out by combining the existing properties and innovative properties. Topics in each network are put in the order according to significance.

Based on above studies, we conclude that consumer community network is the important place that reflects product experiences and facilitates product innovation in future. Manufacturers can promote improvement and innovation of products by exploring effective information on the consumer community network, thus improving the experience level of consumers. On this basis, three strategies to improve information mining in consumer community networks are proposed:

(1) Problems that users concerned are recognized by deep exploring and full understanding of post contents and themes as well as characteristics of cell phone. Problems could be classified reasonably (community division in the network) and core problems could be recognized by multiplex network, thus enabling to solve and guide users' problems in time.

(2) The IUG shall be encouraged and guided to improve the overall interaction performance in the community network. By analyzing the member structure of consumer community, the role of IUG as the bridge between OUG and EUG deserves attention. Enterprises encourage the IUG to interact with OUG and help them to solve problems. This can not only relieve pressure of enterprises in early counseling and late after-sale services but also guide users to improve self-management.

Moreover, enterprise group users shall make use of the key role of IUG in development and test of new products, collecting effective feedbacks quickly and shortening the launch time of new products.

Appendix

A. The Frequency of Initial Tocpics

See Table 7.

B. The Statistics of Valid Posts of Corresponding User and Group

See Table 8.

C. Norm Feature of Z Distribution

Here, we, respectively, build null models of OUG network, IUG network, and EUG network, according to which 1000 random graphs are generated. And Z statistic is built for general variance of degrees of network, which is approximated to normal distribution proved by Kolmogorov-Smirnov test method, so that the "leader" network is determined by "$3-\sigma$" boundary Mar.

Firstly, Z statistics of general variance of degrees of network is defined as follows:

$$Z \triangleq \frac{GVar\,(G)^{zero} - GE_x\,(G)}{\sigma\left[GVar\,(G)^{zero}\right]} \tag{C.1}$$

Table 7: Topics frequency.

Serial number	Topics	Frequency	Serial number	Topics	Frequency	Serial number	Topics	Frequency	Serial number	Topics	Frequency
1	System	6569	26	Game	2810	51	Withdraw money	22	76	Motherboard	183
2	Life	1408	27	Light	692	52	Defrayment	1218	77	Touch	97
3	Taking pictures	3162	28	Flash back	695	53	King Glory	1471	78	Translation	42
4	Hibernation	204	29	Record	216	54	Beta	425	79	Clipboard	3
5	Power consumption	2905	30	Noise	123	55	Jingdong	601	80	Power saving mode	318
6	Data	1780	31	Font	499	56	Vmall	468	81	Wall paper	469
7	Wifi	1992	32	Loudspeaker	130	57	Theme	850	82	3D paper	15
8	Upgrade	4133	33	Program	1012	58	Anti-fingerprint oleophobic coating	622	83	Firmware bag	54
9	Charging	2459	34	Code	1007	59	Ring	472	84	Desktop	1066
10	Power off	934	35	Cloud service	191	60	Battery	2094	85	Radio	75
11	Unlock	1795	36	Lock screen	1968	61	Navigation key	236	86	NFC	622
12	WeChat	5569	37	Memory	1946	62	System B172	228	87	Developer options	129
13	Video	2574	38	Pattern	2684	63	System B213	231	88	Automatic rotation	56
14	Clarity	83	39	Backups	421	64	System B167	269	89	Vague	975
15	Telephone	2806	40	Resolution	398	65	Home key	495	90	Root	432
16	Net	1946	41	Touch off	2	66	Android8.0	95	91	Log version	6
17	Pixel	349	42	Pick-up hand	1719	67	Calendar	117	92	Kugou	54
18	Fingerprint	3241	43	Update	6422	68	Voice assistant	454	93	Compatibility	59
19	Payment	219	44	Camera	1778	69	Type	217	94	Gps	330
20	Block	172	45	Lightness	781	70	Infrared remote control	46	95	Weibo	709
21	Bank card	139	46	Player	137	71	Gesture operation	17	96	QQ	2253
22	Transportation card	345	47	Music	841	72	Application Treasure	34	97	Position	803
23	Consumption	790	48	Screen capture	618	73	Split screen	216	98	Internet speed	443
24	Message	1075	49	Color	774	74	Heat	2700	99	4G	2277
25	Black screen	591	50	Location	600	75	Chip	279	100	Dominant frequency	33

Table 8: Related statistics of valid posts of corresponding user.

Group name	Level name	Number of posts	Frequency of the same group	Overall frequency
OUG	Newcome	18556	24.14%	23.94%
	Beginners	12562	16.34%	16.21%
	Preliminary learners	17743	23.08%	22.90%
	Small success	12578	16.36%	16.23%
	Further progress	7546	9.81%	9.74%
	Master	4591	5.97%	5.92%
	The dedicated	2575	3.35%	3.32%
	The self-contained	385	0.50%	0.50%
	Great success	291	0.38%	0.38%
	Pinnacle	47	0.06%	0.06%
	Magic master	8	0.01%	0.01%
	The matchless	1	0.00%	0.00%
	Limited member	1	0.00%	0.00%
IUG	Hot fans	1	0.29%	0.00%
	Expert fans	81	23.89%	0.10%
	Female fans	1	0.29%	0.00%
	Internal manager	2	0.59%	0.00%
	Internal expert	16	4.72%	0.02%
	Pollen director of city	9	2.65%	0.01%
	Pollen director of universities	1	0.29%	0.00%
	Special forces of HPC	21	6.19%	0.03%
	Moderator of HPC	117	34.51%	0.15%
	Moderator of Huawei Pollen Sub-club	8	2.36%	0.01%
	Moderator of game center	1	0.29%	0.00%
	HRT team	79	28.94%	0.17%
	Super-circle director of HPC	1	0.37%	0.00%
	Theme fans	1	0.37%	0.01%
EUG	Pollen group	134	49.08%	0.00%
	HPC team	11	4.03%	0.10%
	Official team	111	40.66%	0.14%
	EMIUI product manager	3	1.10%	0.00%
	EMIUI official team	5	1.83%	0.01%
	Product manager	9	3.30%	0.01%

where $GVar(G)^{zero}$ is a random variable of general variance of degrees of random network, with $\sigma[GVar(G)^{zero}]$ standard deviation of $GVar(G)^{zero}$ and $GE_x(G)$ the average value of $GVar(G)^{zero}$.

We calculate the general variance of degrees of 1000 random graphs of the OUG network, IUG network, and EUG network. Figures 7–9 show the frequency distribution histogram.

In Figures 7–9, lines represent the normal distribution curve fitting according to the mean and standard deviation of frequency of general variance of degrees of network. And histograms of the three groups of corresponding null models are bell shaped. In order to test whether the distribution of general variance of degrees of null models conform to normal distribution, the K-S test method results are as in Table 9.

From Table 9, we can see that considering norm curve fitting of general variance of degrees' frequency, their Kolmogorov-Smirnov values are all significantly greater than 0.05, indicating that the distribution of general variance of degrees of random networks can approximate normal distribution; that is, $Z \sim N(0,1)$. Moreover, the "3σ" boundary of G denotes Mar (G), which can be calculated through the following form:

Table 9: Kolmogorov-Smirnov test.

Group name	Kolmogorov-Smirnov value	P value
OUG	1.202	0.111
IUG	0.927	0.356
EUG	1.019	0.250

$$\text{Mar}(G) = E\left[GVar(G)^{zero}\right] + 3\sigma\left[GVar(G)^{zero}\right] \quad (C.2)$$

Table 10: The importance ranking and eigenvector corresponding to maximum eigenvalue of three networks.

Hot topics	Eigenvector of OUG	Importance ranking of topics of OUG	Eigenvector of IUG)	Importance ranking of topics of IUG	Eigenvector of EUG	Importance ranking of topics of EUG
System	7.97%	1	4.66%	1	5.62%	2
Life	2.18%	19	0.19%	46	2.65%	13
Taking pictures	2.89%	9	0.81%	35	5.12%	3
Power consumption	3.87%	5	2.87%	15	1.20%	32
Data	1.89%	21	4.26%	2	4.55%	5
Wifi	2.31%	14	3.72%	10	0.91%	38
Upgrade	4.73%	4	1.15%	32	3.59%	8
Charging	2.29%	16	0.38%	42	2.35%	15
Power off	0.93%	30	2.13%	24	0.52%	47
Unlock	1.96%	20	4.12%	6	1.31%	30
WeChat	5.52%	3	0.76%	37	1.54%	27
Video	2.75%	10	0.18%	47	3.67%	7
Telephone	2.30%	15	1.88%	26	1.55%	26
Net	2.24%	17	2.34%	20	3.29%	10
Fingerprint	3.17%	8	3.56%	11	2.31%	16
Transportation card	0.12%	61	0.04%	54	0.03%	58
Consumption	0.74%	38	0.30%	43	3.35%	9
Message	1.43%	28	0.06%	50	1.46%	29
Black screen	0.66%	41	0.01%	57	0.06%	55
Game	3.52%	7	3.42%	12	1.06%	34
Light	0.66%	42	0.26%	45	0.56%	46
Flash back	0.72%	39	1.12%	33	0.00%	59
Font	0.32%	55	0.07%	49	0.08%	54
Program	1.26%	29	3.00%	14	1.71%	23
Code	0.86%	33	3.31%	13	0.93%	36
Lock screen	1.85%	22	2.72%	17	1.74%	22
Memory	1.59%	26	2.60%	18	1.97%	19
Pattern	2.64%	12	4.22%	3	5.71%	1
Backups	0.40%	52	3.95%	9	1.88%	20
Resolution	0.60%	44	2.52%	19	1.09%	33
Pick-up hand	1.51%	27	0.70%	39	4.81%	4
Update	6.64%	2	4.04%	7	1.75%	21
Camera	1.73%	23	0.27%	44	4.09%	6
Lightness	0.88%	32	1.91%	25	1.97%	18
Music	0.78%	36	0.77%	36	1.49%	28
Screen capture	0.44%	49	2.31%	21	0.45%	48
Color	0.61%	43	2.21%	23	3.10%	11
Location	0.70%	40	1.30%	29	2.09%	17
Defrayment	1.63%	25	1.31%	28	1.70%	25
King Glory	1.64%	24	0.50%	40	0.00%	59
Beta	0.32%	54	0.00%	60	0.04%	57
Jingdong	0.43%	50	0.01%	58	0.27%	51
Vmall	0.24%	60	0.01%	58	0.22%	52
Theme	0.43%	51	1.16%	31	0.57%	45
Anti-fingerprint oleophobic coating	0.49%	47	0.05%	52	0.00%	59

Table 10: Continued.

Hot topics	Eigenvector of OUG	Importance ranking of topics of OUG	Eigenvector of IUG)	Importance ranking of topics of IUG	Eigenvector of EUG	Importance ranking of topics of EUG
Ring	0.32%	56	0.05%	51	0.06%	56
Battery	2.64%	11	2.25%	22	2.63%	14
Home key	0.50%	46	0.42%	41	0.21%	53
Voice assistant	0.28%	58	1.83%	27	0.80%	40
Heat	3.54%	6	0.01%	55	0.66%	43
Wall paper	0.28%	59	0.12%	48	0.31%	50
Desktop	0.81%	35	4.17%	4	0.73%	42
NFC	0.36%	53	1.22%	30	0.63%	44
Vague	0.90%	31	0.04%	53	0.92%	37
Root	0.31%	57	4.02%	8	0.99%	35
GPS	0.45%	48	0.00%	61	1.29%	31
Weibo	0.82%	34	1.02%	34	0.80%	41
QQ	2.50%	13	0.74%	38	0.35%	49
Position	0.74%	37	4.16%	5	1.70%	24
Internet speed	0.52%	45	0.01%	56	0.85%	39
4G	2.18%	18	2.80%	16	2.76%	12

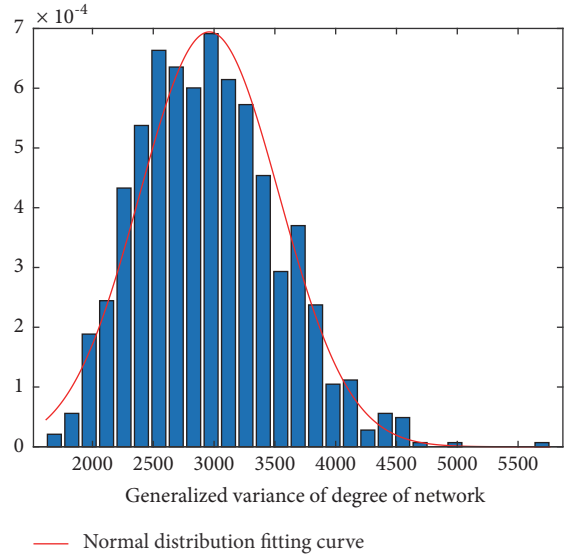


Figure 7: Generalized variance of degree of network of null model of OUG.

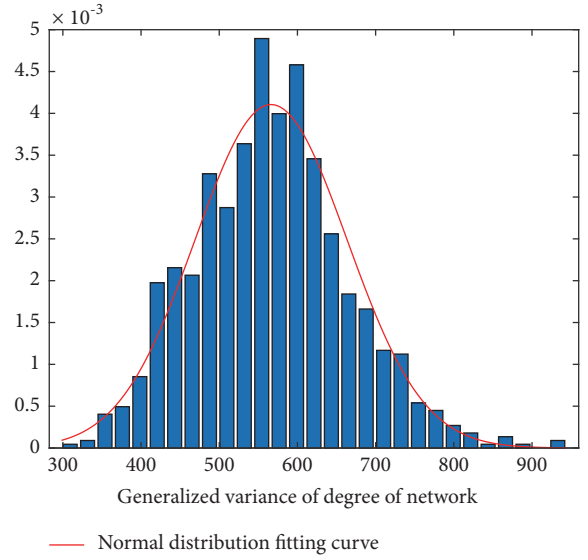


Figure 8: Generalized variance of degree of network of null model of IUG.

Based on the "3σ" principle in statistics, there is a significant difference between G and the corresponding null model, if $GVar(G) > Mar(G)$. As a result, the network G is a "leader" network with uneven importance nodes.

D. The Importance of Topics in Three Networks

See Table 10.

E. The Importance of Topics in Multiplex Netwok

See Table 11.

Data Availability

The Huawei P10/P10 Plus data used to support the findings of this study are available from the corresponding author upon request.

Table 11: The importance and its ranking of topics based on multiplex network.

Hot topics	Importance of topics in multiplex network	Ranking of topics importance
System	18.25%	1
Memory	12.57%	2
Pick-up hand	12.43%	3
Data	10.70%	4
Upgrade	9.46%	5
Pixel	9.04%	6
Taking pictures	8.81%	7
Black screen	8.00%	8
Power consumption	7.95%	9
Net	7.87%	10
WeChat	7.82%	11
4G	7.74%	12
Ring	7.52%	13
Unlock	7.39%	14
Resolution	7.02%	15
Wifi	6.94%	16
Position	6.60%	17
Video	6.60%	18
Code	6.31%	19
Pattern	6.22%	20
Lock screen	6.16%	21
Update	6.09%	22
Font	5.98%	23
Screen capture	5.93%	24
Telephone	5.72%	25
Wall paper	5.70%	26
Vague	5.31%	27
Program	5.10%	28
Life	5.02%	29
Charging	5.02%	30
Camera	4.76%	31
Location	4.64%	32
Transportation card	4.38%	33
Voice assistant	4.21%	34
Backups	4.20%	35
Color	4.10%	36
QQ	3.59%	37
Power off	3.58%	38
Music	3.20%	39
Lightness	3.03%	40
Consumption	2.96%	41
Home key	2.91%	42
Weibo	2.64%	43
Desktop	2.21%	44
Vmall	2.15%	45
Defrayment	2.14%	46
NFC	1.86%	47
Light	1.84%	48
Root	1.75%	49
Game	1.49%	50
Internet speed	1.38%	51
Battery	1.13%	52
Message	0.73%	53
Beta	0.71%	54
Heat	0.70%	55
Theme	0.54%	56
Flash back	0.47%	57
Jingdong	0.46%	58
Anti-fingerprint oleophobic coating	0.42%	59
King Glory	0.36%	60
Fingerprint	0.19%	61

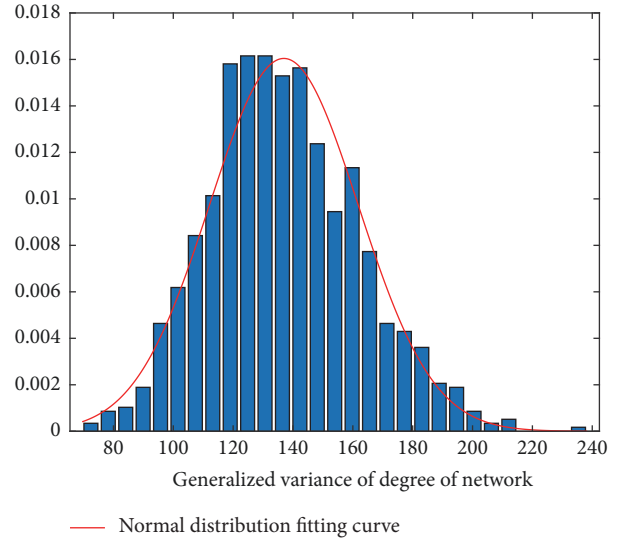


Figure 9: Generalized variance of degree of network of null model of EUG.

Disclosure

Qingchun Meng and Zhen Zhang are co-first authors on this work.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Authors' Contributions

Qingchun Meng and Zhen Zhang contributed equally to this work.

Acknowledgments

This study is supported by the National Nature Science Foundation of China (NSFC) (Grant no. 71572096).

References

[1] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D. W. Hwang, "Complex networks: Structure and dynamics," *Physics Reports*, vol. 424, no. 4-5, pp. 175–308, 2006.

[2] M. E. J. Newman, "The structure and function of complex networks," *SIAM Review*, vol. 45, no. 2, pp. 167–256, 2003.

[3] M. E. J. Newman, *Networks: An Introduction*, Oxford University Press, Oxford, UK, 2010.

[4] S. Dealy, S. Kauffman, and J. Socolar, "Modeling pathways of differentiation in genetic regulatory networks with Boolean networks," *Complexity*, vol. 11, no. 1, pp. 52–60, 2005.

[5] T. Toulouse, P. Ao, I. Shmulevich, and S. Kauffman, "Noise in a small genetic circuit that undergoes bifurcation," *Complexity*, vol. 11, no. 1, pp. 45–51, 2005.

[6] M. Gherardi and P. Rotondo, "Measuring logic complexity can guide pattern discovery in empirical systems," *Complexity*, vol. 21, no. S2, pp. 397–408, 2016.

[7] C. Castellano, S. Fortunato, and V. Loreto, "Statistical physics of social dynamics," *Reviews of Modern Physics*, vol. 81, no. 2, pp. 591–646, 2009.

[8] D. Centola, "The spread of behavior in an online social network experiment," *Science*, vol. 329, no. 5996, pp. 1194–1197, 2010.

[9] K. Lewis, J. Kaufman, M. Gonzalez, A. Wimmer, and N. Christakis, "Tastes, ties, and time: A new social network dataset using Facebook.com," *Social Networks*, vol. 30, no. 4, pp. 330–342, 2008.

[10] S. G. B. Roberts, R. I. M. Dunbar, T. V. Pollet, and T. Kuppens, "Exploring variation in active network size: Constraints and ego characteristics," *Social Networks*, vol. 31, no. 2, pp. 138–146, 2009.

[11] Y. E. Riyanto and Y. X. W. Jonathan, "Directed trust and trustworthiness in a social network: An experimental investigation," *Journal of Economic Behavior Organization*, vol. 151, pp. 234–253, 2018.

[12] J. Shim and J. Kim, "Estimating country-level social network density and supportive surroundings by simulation," *Journal of Business Venturing Insights*, vol. 9, pp. 24–31, 2018.

[13] N. Sumith, B. Annappa, and S. Bhattacharya, "Social network pruning for building optimal social network: A user perspective," *Knowledge-Based Systems*, vol. 117, pp. 101–110, 2017.

[14] J. F. Houston, J. Lee, and F. Suntheim, "Social networks in the global banking sector," *Journal of Accounting and Economics*, 2017.

[15] Z. Qiu and H. Shen, "User clustering in a dynamic social network topic model for short text streams," *Information Sciences*, vol. 414, pp. 102–116, 2017.

[16] M. Fu, H. Yang, J. Feng et al., "Preferential information dynamics model for online social networks," *Physica A: Statistical Mechanics and its Applications*, vol. 506, pp. 993–1005, 2018.

[17] A. Röper, B. Völker, and H. Flap, "Social networks and getting a home: Do contacts matter?" *Social Networks*, vol. 31, no. 1, pp. 40–51, 2009.

[18] Y. Hu, M. Aiello, and C. Hu, "Information diffusion in online social networks: A compilation," *Journal of Computational Science*, vol. 28, pp. 204-205, 2018.

[19] N. Zhang and W. Zhao, "Privacy-preserving data mining systems," *The Computer Journal*, vol. 40, no. 4, pp. 52–58, 2007.

[20] A. Cetto, M. Klier, A. Richter, and J. F. Zolitschka, ""Thanks for sharing"—Identifying users' roles based on knowledge contribution in Enterprise Social Networks," *Computer Networks*, vol. 135, pp. 275–288, 2018.

[21] H. K. Crabb, J. L. Allen, J. M. Devlin, S. M. Firestone, M. A. Stevenson, and J. R. Gilkerson, "The use of social network analysis to examine the transmission of Salmonella spp. within a vertically integrated broiler enterprise," *Food Microbiology*, vol. 71, pp. 73–81, 2018.

[22] B. Simpson and T. McGrimmon, "Trust and embedded markets: A multi-method investigation of consumer transactions," *Social Networks*, vol. 30, no. 1, pp. 1–15, 2008.

[23] X. L. Wan, Z. Zhang, X. X. Rong, and Q. C. Meng, "Exploring an interactive value-adding data-driven model of consumer electronics supply chain based on least squares support vector machine," *Scientific Programming*, vol. 2016, no. 8, Article ID 3717650, p. 13, 2016.

[24] A. Garas and A. Lapatinas, "The role of consumer networks in firms' multi-characteristics competition and market share inequality," *Structural Change and Economic Dynamics*, vol. 43, pp. 76–86, 2017.

[25] J. Ortiz, W.-H. Chih, and F.-S. Tsai, "Information privacy, consumer alienation, and lurking behavior in social networking sites," *Computers in Human Behavior*, vol. 80, pp. 143–157, 2018.

[26] M. Alonso-Dos-Santos, F. Rejón Guardia, C. Pérez Campos, F. Calabuig-Moreno, and Y. J. Ko, "Engagement in sports virtual brand communities," *Journal of Business Research*, vol. 89, pp. 273–279, 2018.

[27] Y. Xiong, Z. Cheng, E. Liang, and Y. Wu, "Accumulation mechanism of opinion leaders' social interaction ties in virtual communities: Empirical evidence from China," *Computers in Human Behavior*, vol. 82, pp. 81–93, 2018.

[28] J. C. Healy and P. McDonagh, "Consumer roles in brand culture and value co-creation in virtual communities," *Journal of Business Research*, vol. 66, no. 9, pp. 1528–1540, 2013.

[29] M. Katz, R. M. Ward, and B. Heere, "Explaining attendance through the brand community triad: Integrating network theory and team identification," *Sport Management Review*, vol. 21, no. 2, pp. 176–188, 2017.

[30] Anonymous, "Research on the product marketing strategies of clothing e-commercial enterprises based on customer experience," *China Textile Leader*, vol. 11, pp. 89–91, 2014.

[31] M.-H. Hsiao and L.-C. Chen, "Smart phone demand: An empirical study on the relationships between phone handset, Internet access and mobile services," *Telematics and Informatics*, vol. 32, no. 1, pp. 158–168, 2014.

[32] A. Paiano, G. Lagioia, and A. Cataldo, "A critical analysis of the sustainability of mobile phone use," *Resources, Conservation & Recycling*, vol. 73, pp. 162–171, 2013.

[33] C.-H. Wang, "Incorporating the concept of systematic innovation into quality function deployment for developing multifunctional smart phones," *Computers & Industrial Engineering*, vol. 107, pp. 367–375, 2017.

[34] IDC, *IDC: Smartphone Vendor Market Share*, IDC, 2018, https://www.idc.com/promo/smartphone-market-share/vendor.

[35] Huawei Pollen Club, *Huawei Pollen Club Forum: Huawei P10/P10 Plus Area*, 2018, https://cn.club.vmall.com/forum-2827-1.html.

[36] Huawei Pollen Club, *Huawei Pollen Club Forum: Huawei P20 Area*, 2018, https://cn.club.vmall.com/forum-3350-1.html.

[37] Huawei Pollen Club, *Huawei Pollen Club Forum: Huawei Mate 10 Area*, Huawei Pollen Club, 2018, https://cn.club.vmall.com/forum-3065-1.html.

[38] N. Hajli, M. Shanmugam, S. Papagiannidis, D. Zahay, and M.-O. Richard, "Branding co-creation with members of online brand communities," *Journal of Business Research*, vol. 70, pp. 136–144, 2017.

[39] D. Georgi and M. Mink, "ECCIq: The quality of electronic customer-to-customer interaction," *Journal of Retailing and Consumer Services*, vol. 20, no. 1, pp. 11–19, 2013.

[40] R. Smaliukiene, L. Chi-Shiun, and I. Sizovaite, "Consumer value co-creation in online business: the case of global travel services," *Journal of Business Economics and Management*, vol. 16, no. 2, pp. 325–339, 2015.

[41] M. Bruhn, S. Schnebelen, and D. Schäfer, "Antecedents and consequences of the quality of e-customer-to-customer interactions in B2B brand communities," *Industrial Marketing Management*, vol. 43, no. 1, pp. 164–176, 2014.

[42] C. Llinares Millán, D. Garzon, and S. Navarro, "C2C interactions creating value in the Route of Santiago," *Journal of Business Research*, vol. 69, no. 11, pp. 5448–5455, 2016.

[43] W. Wei, Y. Lu, L. Miao, L. A. Cai, and C.-Y. Wang, "Customer-customer interactions (CCIs) at conferences: An identity approach," *Tourism Management*, vol. 59, pp. 154–170, 2017.

[44] Y. Chen and H. Zhang, "The impact of customer to customer interaction on service company-customer relationship quality," in *Proceedings of the 8th International Conference on Service Systems and Service Management, ICSSSM'11*, vol. 11, China, June 2011.

[45] J. S. Oh, J. I. Shin, D. Y. Jeong, and K. H. Chung, "The effects of message directionality and brand image's level on consumer attitude," *ICIC Express Letters, Part B: Applications*, vol. 6, no. 4, pp. 1003–1008, 2015.

[46] R. Zollet and A. Back, "Critical factors influencing diffusion of interactivity innovations on corporate websites," *IEEE Transactions on Professional Communication*, vol. 58, no. 1, pp. 2–19, 2015.

[47] I. Khan, H. Dongping, and A. Wahab, "Does culture matter in effectiveness of social media marketing strategy? An investigation of brand fan pages," *Aslib Journal of Information Management*, vol. 68, no. 6, pp. 694–715, 2016.

[48] H. Nourikhah and M. K. Akbari, "Impact of service quality on user satisfaction: Modeling and estimating distribution of quality of experience using Bayesian data analysis," *Electronic Commerce Research and Applications*, vol. 17, pp. 112–122, 2016.

[49] S. Shobeiri, E. Mazaheri, and M. Laroche, "How customers respond to the assistive intent of an E-retailer?" *International Journal of Retail & Distribution Management*, vol. 42, no. 5, pp. 369–389, 2014.

[50] D.-K. Liou, L.-C. Hsu, and W.-H. Chih, "Understanding broadband television users' continuance intention to use," *Industrial Management & Data Systems*, vol. 115, no. 2, pp. 210–234, 2015.

[51] J. Islam and Z. Rahman, "The impact of online brand community characteristics on customer engagement: An application of Stimulus-Organism-Response paradigm," *Telematics and Informatics*, vol. 34, no. 4, pp. 96–109, 2017.

[52] M. Kilgour, S. L. Sasser, and R. Larke, "The social media transformation process: Curating content into strategy," *Corporate Communications*, vol. 20, no. 3, pp. 326–343, 2015.

[53] S. McKechnie and P. Nath, "Effects of new-to-market e-store features on first time browsers," *International Journal of Human-Computer Studies*, vol. 90, pp. 14–26, 2016.

[54] K. Chu, D. Lee, G. Y. Kim, and J. Y. Kim, "Not All Communication is Created Equal : An Investigation into the Way to Improve Communication Quality on SNS," *JOurnal of Korean Marketing Association*, vol. 32, no. 2, pp. 103–124, 2017.

[55] C.-H. Chiang and J.-C. Wang, "The impact of interaction networks on lurkers' behavior in online community," in *Proceedings of the 48th Annual Hawaii International Conference on System Sciences, HICSS 2015*, pp. 1645–1656, USA, January 2015.

[56] Q. Li and J. Gu, "Activity driven modelling of online social network," *Journal of Systems Engineering*, vol. 30, pp. 9–15, 2015.

[57] R. Andersen and A. I. Mørch, "Mutual development in mass collaboration: Identifying interaction patterns in customer-initiated software product development," *Computers in Human Behavior*, vol. 65, pp. 77–91, 2016.

[58] P. Baumgartner and N. Peiper, "Utilizing Big Data and Twitter to Discover Emergent Online Communities of Cannabis Users," *Substance Abuse: Research and Treatment*, vol. 11, 2017.

[59] X. J. Liu, Z. Narisa, and X. L. Cui, "Construction and application of the complex network about consumer online reviews based on latent Dirichlet allocation model," *Systems Engineering*, vol. 32, no. 3, pp. 305–312, 2017.

[60] P. V. Marsden, "Interviewer effects in measuring network size using a single name generator," *Social Networks*, vol. 25, no. 1, pp. 1–16, 2003.

[61] H. R. Bernard, P. Killworth, D. Kronenfeld, and L. Sailer, "The problem of informant accuracy: the validity of retrospective data.," *Annual Review of Anthropology*, pp. 495–517, 1984.

[62] C. T. Butts, "Network inference, error, and informant (in)accuracy: A Bayesian approach," *Social Networks*, vol. 25, no. 2, pp. 103–140, 2003.

[63] Huawei Pollen Club, *Application Center: Application of Technical Talent*, 2017, https://cn.club.vmall.com/thread-11749795-1-1.html.

[64] Huawei Pollen Club, *Application Center: Application of Management Talent*, 2017, https://cn.club.vmall.com/thread-11749629-1-1.html.

[65] Huawei Pollen Club, *Application Center: Application of Pollen Talent*, 2017, https://cn.club.vmall.com/thread-11748231-1-1.html.

[66] Huawei Pollen Club, *Application Center: Application of Pollen Girl*, 2017, https://cn.club.vmall.com/thread-11747691-1-1.html.

[67] Huawei Pollen Club, *Application Center: Rules of Area Management*, 2017, https://cn.club.vmall.com/thread-11747831-1-1.html.

[68] Huawei Pollen Club, *Application Center: Application Form and Format of Pollen Club*, 2017, https://cn.club.vmall.com/thread-13295938-1-1.html.

[69] G. R. Chen, X. F. Wang, and X. Li, *Introduction to complex network: Models, structures and dynamics*, Higher Education Press, Beijing, China, 2012.

[70] V. D. Blondel, J. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2008, no. 10, Article ID P10008, pp. 155–168, 2008.

[71] C. J. Liu, *Community ditection and analytical application in complex networks [Dissertation, thesis]*, Shandong University, 2014.

[72] J. Fu, J. Wu, C. Liu, and J. Xu, "Leaders in communities of real-world networks," *Physica A: Statistical Mechanics and its Applications*, vol. 444, pp. 428–441, 2016.

[73] J. Iranzo, J. M. Buldú, and J. Aguirre, "Competition among networks highlights the power of the weak," *Nature Communications*, vol. 7, p. 13273, 2016.

[74] S. Boccaletti, G. Bianconi, and R. Criado, "The structure and dynamics of multilayer networks," *Physics Reports*, vol. 544, no. 1, pp. 1–122, 2014.