UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

POSGRADO EN CIENCIA E INGENIERÍA DE LA COMPUTACIÓN

RANDOM BOOLEAN NETWORKS, STUDY OF DEGENERACY

PRESENTA:
ROBERTO GUTIÉRREZ GARCÍA

T E S I S
QUE PARA OPTAR POR EL GRADO DE:
MAESTRO EN CIENCIAS (COMPUTACIÓN)

DR. DAVID A. ROSENBLUETH LAGUETTE Y DR. CARLOS GERSHENSON GARCÍA
IIMAS - UNAM
PROGRAMA DE POSGRADO EN CIENCIA E INGENIERÍA DE LA COMPUTACIÓN

MÉXICO, D.F. SEPTIEMBRE 2013

Thesis Advisor Dr. David A. Rosenblueth Laguette

Thesis Reader Dr. León Patricio Martínez Castilla

Thesis Advisor Dr. Carlos Gershenson García

Thesis Reader Dr. Maximino Aldana González

Chairman of Department Dr. Fernando Arámbula Cosio

Thesis Reader Dr. Pedro Miramontes Vidal

Random Boolean Networks, Study of Degeneracy

By

Roberto Gutiérrez García

Dissertation

Submitted in Partial Fulfillment of the Requirements
for the Degree of Master
in Computer Sciences
in the "Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas" IIMAS at
National Autonomous University of Mexico UNAM, 2013

Mexico City, Mexico

"At the center of the problem is the process of the self-organization of matter."

Albert L. Lehninger

"The world is either the effect of cause or chance. If the latter, it is a world for all that, that is to say, it is a regular and beautiful structure."

Marcus Aurelius

A mis padres, Amada y Max, por su amor y apoyo incondicional.

A Francisco y María Fernanda, por su gran cariño y compañia

Agradecimientos

En primer lugar, agradezco a mi familia. A mis padres, Maximino y Amada, por el amor que me han brindado toda la vida. A mi hermano Francisco, con quien he compartido mi vida desde mis primeros años. A María Fernanda, mi sobrina (hermana), por esas tardes en las me convierto de vuelta en niño, mostrandome lo hermosa y sencilla que es la vida. A mis demás tíos, primos y toda la familia, que han estado al pendiente de mi

Un especial agradecimiento a mis tíos Fabiana y Severo, por el invaluable apoyo y cariño durante la realización de mi maestría.

Agradezco a mi tutor Carlos, a quien admiro no sólo por su inteligencia, ideas y trabajo, sino por su calidad humana.

A todos mis amigos, por sus locuras, kilómetros recorridos, los amenos momentos, y por su apoyo incondicional.

El desarrollo de esta tesis fue apoyado por el CONACyT beca 371.369 y PAEP, UNAM.

Abstract

Random Boolean networks (RBNs) are general models of gene regulatory networks that can be used to explore evolution theories. Understanding the relationship between robustness and evolvability is key to explaining how living systems can withstand mutations, while produce variation that may lead to evolutionary innovations. Here we explore the hypothesis that degeneracy plays a central role in robustness and thus evolvability. We propose three types of degeneracy for RBNs. Results suggest that all three types of degeneracy increase the robustness of RBNs and decrease their attractor lengths. Our results complement the evidence in favor of degeneracy as a promoter of evolvability.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Life exhibits many interesting features which are difficult to create artificially. At almost all biological organization scales, we see that systems are versatile and robust to changing conditions [WJM10]. Thus, many phenomena can be described as *complex adaptive systems* (CAS). CAS are *complex* in the sense that they are conformed by multiple interconnected elements, and *adaptive*, because these systems have the capacity to change their functionality. The individual and collective behaviors change and self-organizes [Mit09].

Common examples of CAS are the immune system, stock markets, gene regulatory networks, social networks, Internet, etc. These systems consist of a large number of components, that interact with each other. Emergent behaviors and properties arise at the system level. A system is robust if it can maintain its function in face of perturbations [Wag05b].

Robustness is in some sense unique, because it is obtained through a richly distributed response that allows CAS to handle challenging variations of environmental stress [WJM10].

Although robust, biological systems sometimes can adapt themselves to explore new resources and persist in unpredictable environments [Whi10]. Therefore, a second im-

portant feature is the ability to innovate. In other words, *evolvability* is the capacity to discover beneficial, heritable adaptations. [WA96].

Moreover, robustness plays a key role in evolvability allowing the gradual exploration of new solutions, while maintaining functionality. A small change in a fragile system would destroy it. There are several mechanisms that help a system to be robust [von56], such as modularity [Sim96, WP05], distributed robustness [Wag05a], redundancy, and degeneracy [LB04]. In this work, we focus on the latter.

A deep understanding of CAS requires the comprehension of the conditions which facilitate the coexistence of high robustness, growing complexity, and the continued propensity for innovation, *i.e.* evolvability [Whi10]. "Understanding the relationship between robustness and evolvability is key to understand how living things can withstand mutations, while producing ample variation that leads to evolutionary innovations" [CMW07].

Many CAS can be usefully described as network, where nodes represent elements, and their connections represent their interactions. Network connectivity relates directly to robustness because it allows for mutations and perturbations that do not change the phenotype [Wag05b]. The degree of robustness depends on the local topology and the size of the network. Besides, evolvability is concerned with long-term movements that can reach over widely different regions of the fitness landscape[1]. An extensive neutral network[2] with a rich phenotypic neighborhood allows the exploring of many diverse phenotypes without affecting the core functionalities of a system [WB10a].

[1]The set of all possible genotypes, their degree of similarity, and their related success values is called a fitness landscape.

[2]A neutral network is a set of points in the search space which have the same fitness. Neutral networks are also defined as points in the search space that are connected through neutral point-mutations where the fitness is the same for all the points in such network [Kim83].

Degeneracy

In biology, degeneracy refers to conditions where the functions or capabilities of components partially overlap [Whi10, WB10a]. In other words, degeneracy is the property of having similar results from different pathways or different processes, emerging through the actions of multiple dissimilar parts [Wag05a, Wag05c].

This property is not visible at a single scale because it is present in complex systems where heterogeneous components (e.g. gene products) have multiple interactions with each other. In this sense, degeneracy is a system property that requires the existence of multi-functional components (but also modules and pathways), which have similar functionalities under certain conditions and may play different roles under other conditions [GSFF00b].

Degeneracy is intimately related to complexity because while complex systems are both functionally integrated and functionally segregated, degenerate components are both functionally redundant and functionally independent [Whi10].

In the same sense, Edelman and Gally [EG01] point out that degeneracy is a ubiquitous biological property and argue that it is a feature of complexity at the genetic, cellular, organismic, and population levels. Furthermore, it is both necessary for, and an inevitable outcome of, natural selection, and contributes to robustness because it provides a mechanism to obtain the same functionality through a diversity of components.

The contrast between degeneracy and redundancy at the structural level is sharpened by comparing design and selection in engineering and evolution, respectively. In engineering systems, logic prevails, and, for fail-safe operation, redundancy is built into design. This is not the case for biological systems [EG01].

Modeling degeneracy in RBN

Random Boolean networks (RBN) are appealing models for studying evolution, since functionality and topology are not assumed. RBNs have different dynamic regimes. On the one hand, the dynamics which stabilize quickly, is called *ordered* regime. On the other hand, the dynamics where RBNs are generally sensitive to initial conditions, is called *chaotic* regime. The phase transition from the ordered to the chaotic regime, is known as the *edge of chaos*, where small changes do not destroy previous functionality, but can explore their space of possibilities.

This thesis shows the effects of degeneracy for achieving robustness and evolvability, in face to perturbations, in discrete dynamical systems, using random Boolean networks as model of study.

In this work, three types of degeneracy are defined: function degeneracy, input degeneracy and output degeneracy. Based on this classification, computational experiments are developed to compare the effects different types of degeneracy and redundancy on the dynamics of RBNs. Sensitivity to initial conditions, complexity, and information are also measured. Experiments measuring complexity and information are implemented.

The relevance of this thesis, lies in the study of degeneracy of CAS, particularly in ran-

dom Boolean networks. The relationship between degeneracy, as a form of robustness and ability to innovate (adaptability) of living beings is analized.

This thesis is organized as follows: chapter 2 provides an introduction to discrete dynamical systems. Chapter 3 presents a description of random Boolean networks. Chapter 4 presents a description of how degeneracy is defined and implemented, from duplicate elements (redundant nodes), to elements with a structural and functional partial change. Chapter 5 shows the results obtained and a discussion about experiment outcomes. Chapter 6 concludes the thesis.

Chapter 2

Dynamical networks

There are many different ways to study CAS by means of: agent-based models, complex network-based models, and differential equations. Agent-based models are developed by means of various methods and tools, identifying the different agents inside the model and capturing the essential characteristics and behavior. Another method for modeling CAS involves building complex network models by using interaction data of various CAS components. A different approach for modeling complex systems may be using differential equations, (particular examples are [GK73, KEG02], for GRNs) in which interactions are incorporated as logical functions.

Dynamical networks are interesting models since one does not have to assume any functionality or particular connectivity of networks, to study their generic properties. Dynamical networks are used for exploring the configurations where life could emerge [Ger04a]. In this chapter we discuss the most important concepts of network theory (graph theory) and basic realizations of possible network organizations. The definitions and concepts are based on [Gro08].

2.1 Real-World Networks

Dynamical networks constitute a wide class of CAS. There are plenty of examples in nature, ranging from social insects such as ants, to the biosphere and the ecosystems,

as well as the immune system. The brain of vertebrate and most invertebrate animals, is another example of adaptive dynamical system, composed of interconnected neurons, via a synapse. In the same way, in life we interact through social networks at different scales. Networks are ubiquitous models throughout the domain of living creatures.

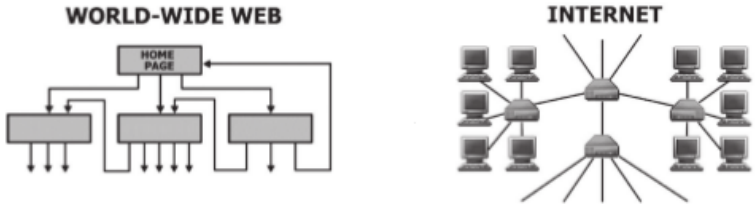


Fig. 2.1: Illustration of the network structure of the world-wide web (left) and of the Internet (right); from [Bar02].

Social networks are an example of a communication network. Most human communication takes place directly among individuals. The spreading of rumors, jokes and diseases takes place by contact between individuals.

Information processing is ubiquitous in networks; well known examples are the Internet and the world-wide web, see Fig. 2.1. Inside a cell the many constituent proteins form an interacting network, as illustrated in Fig. 2.2. The same is of course true for artificial neural networks as well as for the networks of neurons that constitute the human brain. It is therefore important to understand the statistical properties of the most important network classes.

2.2 Graph-Theoretical Concepts

In this section, we introduce some concepts allowing us to characterize theoretical graphs and real-world networks.

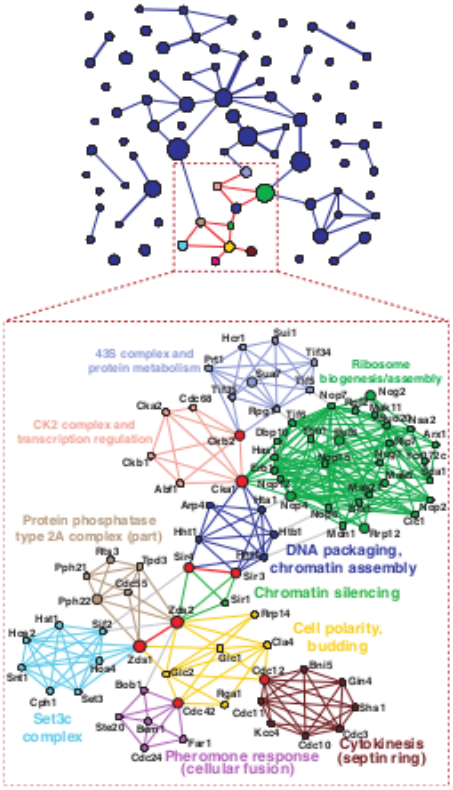


Fig. 2.2: A protein interaction network, showing a complex interplay between highly connected hubs and communities of subgraphs with increased densities of edges from [PDFV05]).

Degree of a Vertex

A graph is made out of vertices connected by edges, determining the degree of a vertex, as defined below:

Degree of a Vertex. The degree k of the vertex is the number of edges linking such a vertex to either another node in the network, or itself.

Nodes having a degree k substantially above the average are denoted *hubs*, which are relevant to the network theory studies.

Coordination Number

The simplest type of network is the random graph. It is characterized by only two numbers: By the number of vertices N and by the average degree z, also called the coordination number.

Coordination Number. The coordination number z is the average number of links per vertex, namely the average degree.

A graph with an average degree z has $Nz/2$ connections. Alternatively we can define with p the probability to find a given edge.

Connection Probability. The probability that an edge between two nodes in the network occurs is called the connection probability p.

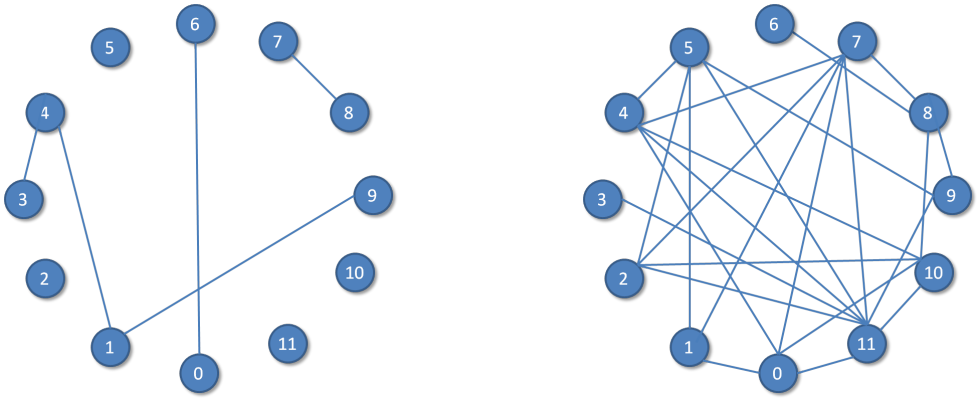


Fig. 2.3: Random graphs with $N = 12$ vertices and different connection probabilities $p = 0.08$ (left) and $p = 0.4$ (right). The three mutually connected vertices (0,1,7) contribute to the clustering coefficient and the fully interconnected set of sites (0,4,10,11) is a clique in the network on the right.

Erdös-Rényi Random Graphs

We can construct a specific type of random graph simply by taking N nodes, also called vertices and by drawing $Nz/2$ lines, the edges, between randomly chosen pairs of nodes, compare Fig. 2.3. This type of random graph is called an *Erdös-Rènyi* random graph.

For Erdös-Rényi random graphs we have

$$p = \frac{Nz}{2}\frac{2}{N(N-1)} = \frac{z}{N-1} \tag{2.1}$$

for the relation between the coordination number z and the connection probability p.

The Thermodynamic Limit

Most systems in nature are not in equilibrium; i.e. they are changing over time, and are continuously and discontinuously subject to flux of matter and energy from other systems and chemical reactions.

Thermodynamic Limit. The *thermodynamic limit*, or *macroscopic limit*, is the large N

limit in statistical mechanics, where N is the number of particles, such as (atoms or molecules), in the system.

A property is extensive if it is proportional to the amount of constituting elements, and intensive if this property does not depend on the system size or the amount of material in the system. We note that $p = p(N) \to 0$ in the thermodynamic limit $N \to \infty$ for Erdös-Rényi random graphs and intensive $z \sim O(N^0)$; compare Eq. 2.1.

There are small and large real-world networks and it makes sense only for large networks to consider the thermodynamic limit. An example is the network of hyperlinks in the world wide web, as shown in Fig. 2.4[1].

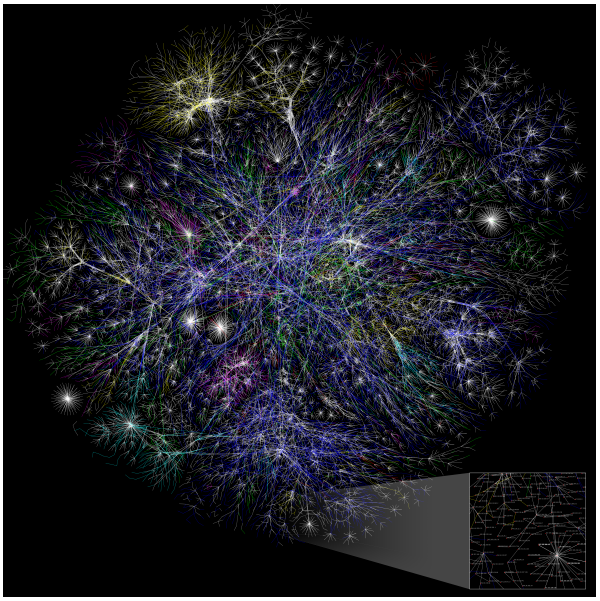


Fig. 2.4: Partial map of the Internet based on the January 15, 2005. Each line is drawn between two nodes, representing two IP addresses. This graph represents less than 30% of the Class C networks reachable by the data collection program in early 2005.

[1]Image by The Opte Project, data can be found on opte.org

Chapter 3

Random Boolean Networks

Complex systems theory deals with dynamical systems containing a large number of variables. The resulting dynamical behavior can be arbitrarily complex and sophisticated. For measuring the properties of dynamical systems is therefore important to consider the scale at which experiments are performed, as small networks can show a different behavior than large-scale networks.

Networks of interacting binary variables, *i.e.* Boolean networks, constitute a canonical complex dynamical systems for the study of dynamics and many properties of biological systems, exploring them with different topologies.

Random Boolean networks can hence be considered, in a certain sense, as being of prototypical importance in this field, as they provide well defined classes of dynamical systems for which the thermodynamical limit $N \to \infty$ can be taken. They show chaotic as well as regular behavior, despite their apparent simplicity, and many other dynamical systems phenomena. Also, in the thermodynamic limit there can be phase transitions between chaotic and regular regimes.

One of the aims of RBN research is studying how the topology of a network (structure) determines the state transition properties of the network (function), which is the conceptual framework of this study. Note that all the definitions and concepts of this

chapter are from the book [Gro08].

3.1 Introduction

Random Boolean networks (RBNs) were developed by Stuart Kauffman as a model of gene regulatory networks (GRNs) [Kau69, Kau93, Ger04a], which consist of N nodes with a Boolean state, representing whether a gene is being transcribed or not (*on* or *off* respectively). These states are determined by the states of K nodes which can be considered as inputs or links towards a node. Because of this, RBNs are also known as NK networks or Kauffman models [AGCK03]. The states of nodes are updated by lookup tables that specify for every 2^K possible combination of input states the future state of the node.

RBNs are random in the sense that the connectivity (which nodes are inputs of which, see Fig. 3.1) and functionality (lookup tables of each node, see Table 3.1) are chosen randomly when a network is generated, although these remain fixed as the network is updated each time step. RBNs are discrete dynamical networks (DDNs), since they have discrete values, number of states, and time [Wue98]. They can also be seen as a generalization of Boolean cellular automata [NC97, Ger02], where each node has a different neighborhood and rule.

In RBNs, each state has only one successor, while having the possibility of having several predecessor states (many states lead to one state), or no predecessor (states without predecessors are called *Garden of Eden* states). In this way, transitions between network states determine the state space of the RBN.

In classic RBNs, the updating is deterministic and synchronous [Ger02]. Since they have 2^N possible network states, sooner or later a state will be repeated. When this occurs, the network has reached an attractor, since the dynamics will remain in that subset of the state space. If the attractor consists of only one state, then it is called a *point attractor* (similar to a steady state), whereas an attractor consisting of several states is called a *cycle attractor* (similar to a limit cycle).

According to Kauffman, the number of cell types in an organism is similar to the number of attractors in a RBN, with number of nodes similar to the number of genes of the organism [Kau69]. In this context, all cells of an animal contain the same genes and cell differentiation, i.e. the fact that a skin cell differs from a muscle cell, is because of the differences in the gene activities in the respective cells.

Kauffman's original model has been generalized to a wide spectrum of applications, such as to analyze and predict genomic interactions [SS96, SFAW97], in neural networks [HMAG02], robotics [QNDR03] and music generation [Dor00]. Therefore, Boolean networks are general models to study diverse dynamical systems because we can model at an abstract level many phenomena, and study generic properties of networks independently of their functionality [Ger04a]. For modeling particular genetic networks, other models have been proposed, differing by their updating schemes [HB97].

3.2 Boolean Networks

In this section, we describe the dynamics of a set of N binary variables.

Boolean Variables A Boolean or binary variable has two possible values, we denote

them as 0 and 1 in this work.

These variables interact according to rules denoted as coupling functions.

Boolean Coupling Functions. A Boolean function $\{0,1\}^K \to \{0,1\}$ maps K Boolean variables into a single one.

The dynamics of the system is considered to be discrete, $t = 0, 1, 2, \ldots$. The value of the variables at the next time step is determined by the choice of Boolean coupling functions. The set of all these elements and their linkage is denoted as the Boolean network.

The Boolean network. The set of Boolean coupling functions interconnecting the N Boolean variables can be represented graphically by a directed network, the Boolean network.

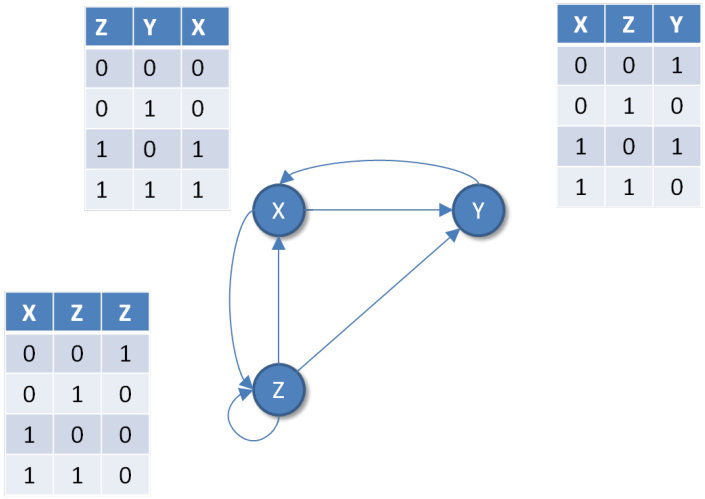


Fig. 3.1: Illustration of a Boolean network with $N = 4$ nodes and a connectivity $K = 2$, and their lookup table.

In Fig. 3.1 a small Boolean network of size $N = 4$ is illustrated.

3.3 Random Variables and Networks

Modeling real-world systems by a collection of interacting binary variables is often a simplification, as real-world are often represented by continuous variables. For the case of the gene expression network, we just keeps two possible states for every single gene: *active* or *inactive*.

Thresholds are parameter regimes at which the dynamical behavior changes qualitatively, and are wide-spread in biological systems. Examples are neurons, which fire or do not fire depending on the total strength of presynaptic activity. Similar thresholds occur in metabolic networks in the form of activation potentials for the chemical reactions involved. However, thresholds are not of interest for the purpose of this work.

Boolean networks have a rich variety of possible concrete model realizations, which differ in updating scheme, topology and and type of coupling ensembles.

3.3.1 Boolean Variables and State Space

We denote by

$$\sigma_i \in 0, 1, \qquad i = 1, 2, \ldots, N$$

the N binary variables and by Σ_t the state of the system at time t,

$$\Sigma_t = \{\sigma_1(t), \sigma_2(t), \ldots, \sigma_N(t)\}. \tag{3.1}$$

Σ_t can be thought of as a vector pointing to one of the $\Omega = 2^N$ edges of an N-dimensional hyper-cube, where Ω is the number of possible configurations, in a given time step. Thus,

σ_1	σ_2	σ_3	Random (a)	Canalizing (b)	Additive Gen. (c)	XOR (c)
0	0	0	0	1	0	0
0	0	1	1	1	0	1
0	1	0	1	1	0	1
0	1	1	0	1	1	0
1	0	0	1	0	0	1
1	0	1	0	0	1	0
1	1	0	1	0	1	0
1	1	1	1	0	1	1

Table 3.1: Examples of Boolean functions of three arguments. (a) A particular random function. (b) A canalizing function of the first argument; when $\sigma_1 = 0$, the function value is 1. If $\sigma_1 = 1$, then the output can be either 0 or 1. (c) An additive function. The output is 1 (active) if at least two inputs are active. (d) The generalized XOR, which is true when the number of 1-bits is odd.

.

time is assumed to be discrete,

$$\sigma_i = \sigma_i(t), \qquad t = 1, 2, \ldots .$$

The value of a given Boolean element σ_i at the next time step is given by the values of K controlling variables.

Controlling Elements. The controlling elements $\sigma_{j_1(i)}, \sigma_{j_2(i)}, \ldots, \sigma_{j_K(i)}$ of a Boolean variable σ_i determine its time evolution by

$$\sigma_i(t+1) = f_i(\sigma_{j_1(i)}(t), \sigma_{j_2(i)}(t), \ldots, \sigma_{j_K(i)}(t)). \tag{3.2}$$

Here f_i is a Boolean function associated with σ_i. The set of controlling elements might include σ_i itself. Some exemplary Boolean functions are given in the Table 3.1.

Number of Coupling Functions

The coupling function:

$$f_i : \qquad \{\sigma_{j_1(i)}, \ldots, \sigma_{j_K(i)}\}, \rightarrow \sigma_i \tag{3.3}$$

has 2^K different arguments. To each argument value we can assign either 0 or 1. Thus there are a total of

$$|f_{(K)}| = 2^{(2^K)} = 2^{2^K} \tag{3.4}$$

possible coupling functions. For example, if we have a connectivity $K = 3$, we will have $|f_{(3)}| = 2^{2^3} = 256$ possible functions. In Table 3.1 we present several examples for the mentioned example.

Model Realizations

A given set of linkages and Boolean functions $\{f_i\}$ defines what we call a realization of the model. The dynamics then follows from Eq. 3.2. For the updating of all elements during one time step we have several choices:

- *Synchronous updating:* All variables $\sigma_i(t)$ are updated simultaneously.
- *Serial updating (or asynchronous updating):* Only one variable is updated at every step. This variable may be picked at random or by some predefined ordering scheme.

The updating scheme does not affect thermodynamic properties. However, the occurrence and the properties of cycles and attractors discussed in Sect. 3.5, crucially depends on the updating scheme.

3.3.2 Model Definition

Finally, for a complete definition of the model we specify several parameters:

The Connectivity: The first step is to select the connectivity K_i of each element, i.e. the number of its controlling elements. With

$$\langle K \rangle = \frac{1}{N} \sum_{i=1}^{N} K_i$$

the average connectivity is defined. Here we will consider mostly the case in which the connectivity is the same for all nodes: $K_i = K, i = 1, 2, \ldots, N$.

The connecting input links: The second step is to select the specific set of controlling elements $\sigma j_1(i), \sigma_{j_2(i)}, \ldots, \sigma_{j_K(i)}$ on which the element σ_i depends. See Figure 3.1 for an illustration.

The Evolution Rule: The third step is to choose the Boolean function f_i determining the value of $\sigma_i(t+1)$ from the values of the linkages $\sigma_{j_1(i)}(t), \sigma_{j_2(i)}(t), ..., \sigma_{j_K(i)}(t)$.

Besides, the way the connecting input links are assigned determines the topology of the network. There are diverse topologies. We mention three general cases: lattices, uniform and scale-free topologies. Other type of topology are small-world networks, with regular short-distance links and random long-distance links are popular models in network theory.

3.4 The Dynamics of Boolean Networks

Although living entities follow the laws of physics and chemistry, several key features of biological systems result from the relevance that information plays, are not shared

by physical systems [Gou02]. The transmission, storage and manipulation of information in biological systems let the creation of biological structures through evolutionary pathways which are contingent [Hop94].

Information allows biological systems to compute. More complex organisms are better able to cope with environmental uncertainty because they can compute, i.e. they have memory or some form of internal plasticity, and they can also make calculations that determine the appropriate behavior using what they sense from the outside world [FS04].

In Boolean networks, the dynamics needs to be sufficiently versatile for adaptive behavior but short of chaotic to ensure reliable behavior, and this in turn implies a balance between order and chaos in the network.

There is an emergent property in CAS, a network's ability to dynamically categorize its state-space. The emergent structure that embodies memory is the network's basin of attraction field, representing all possible trajectories through state-space. Categorization occurs far from equilibrium as well as at attractors, creating a complex hierarchy of content addressable memory represented by separate attractors within the attraction field, and also by the root's subtrees in basins of attraction [Wue97]. In RBNs, as well as in many dynamical systems, three regimes can be distinguished: *ordered*, *critical*, and *chaotic* [Ger04a].

3.4.1 The flow of information through the network

For random models which maintain the same updating scheme, the response to change is relevant, since changing the value of any variable σ_i can cause either a small or large

effect in the dynamics of the system. We may either change the initial conditions, or some specific coupling function, and examine its effect on the time evolution of the variable considered.

A GRN, to give an example, for which even small damage routinely results in the death of the cell, will be at an evolutionary disadvantage with respect to a more robust gene expression set-up. Here we will examine the sensitivity of the dynamics with regard to the initial conditions. A system is robust if two similar initial conditions lead to similar long-time behavior. One way to measure the robustness of a system is by the difference in their states, so that we need some definitions:

The Hamming Distance

We consider two different initial states,

$$\sigma_0 = \{\sigma_1(0), \sigma_2(0), \ldots, \sigma_N(0)\}, \qquad \tilde{\sigma}_0 = \{\tilde{\sigma}_1(0), \tilde{\sigma}_2(0), \ldots, \tilde{\sigma}_N(0)\}.$$

Typically we are interested in the case when Σ_0 and $\tilde{\Sigma}_0$ differ in the values of only by one element. A suitable measure for the distance is the *Hamming distance* $D(\sigma, \tilde{\sigma}) \in [0, N]$,

$$D_H(\sigma, \tilde{\sigma}) = \sum_{i=1}^{N} |\sigma_i - \tilde{\sigma}_i|, \tag{3.5}$$

which is just the sum of elements that differ in Σ_0 and $\tilde{\Sigma}_0$. As an example we consider

$$\Sigma_1 = \{1, 0, 0, 1\}, \qquad \Sigma_2 = \{0, 1, 1, 0\}, \qquad \Sigma_3 = \{1, 0, 1, 1\}.$$

We have $D_H = 4$ for the Hamming distance $\Sigma_1 - \Sigma_2$ and $D_H = 1$ for the Hamming

distance $\Sigma_1 - \Sigma_3$. If the system is robust, two close-by initial conditions will never diverge with passing time, in terms of the Hamming distance.

The Normalized Hamming Distance

The normalized Hamming distance $D_n(\sigma, \tilde{\sigma}) \in [0, 1]$ between two configurations is defined as:

$$H(\sigma, \tilde{\sigma}) = \frac{D(\sigma, \tilde{\sigma})}{N} \tag{3.6}$$

3.4.2 Dynamical Regimes

RBNs have three dynamical regimes: *ordered*, *chaotic*, and *critical* [Wue98, Ger04a]. Typical dynamics of the three regimes can be seen in Fig. 3.2. The ordered regime is characterized by small change, *i.e.* most nodes are static. The chaotic regime is characterized by frequent changes, *i.e.* nodes are changing.

RBNs in the ordered regime are robust to perturbations (of states, of connectivity, of node functionality). Since most nodes do not change, damage has a low probability of spreading through the network. By contrast, RBNs in the chaotic regime are fragile: since most nodes are changing, damage spreads easily, creating large avalanches that spread through the network. The critical regime balances the ordered and chaotic properties: the network is robust to damage, but it it not static. This balance has led people to argue that life and computation should be within or near the critical regime [Lan90, Kau93, Cru94, Kau00]. In the ordered regime, there is robustness, but no possibility for dynamics, computation, and exploration of new configurations, *i.e.* evolution. In the chaotic regime, exploration is possible, but the configurations found are fragile,

Fig. 3.2: Trajectories through state space of RBNs within different phases, $N = 32$. A square represents the state of a node. Initial states at top, time flows downwards. *Left:* ordered, $K = 1$. *Center:* critical, $K = 2$. *Right:* chaotic, $K = 5$.

i.e. it is difficult to reach persisting patterns (memory). In Fig. 3.2 we can appreciate characteristic dynamics of RBNs in different phases.

It has been found that the regimes of RBNs depend on several parameters and properties [Gerss]. Still, two of the most salient parameters are the connectivity K and the probability p that there is a 1 on the last column of lookup tables. When $p = 0.5$ there is no probability bias. For $p = 0.5$, the ordered regime is found when $K < 2$, the chaotic regime when $K > 2$, and the critical regime when $K = 2$ [Ger04a]. The ordered and chaotic regimes are found in distinct phases, while the critical regime is found on the phase transition. Derrida and Pomeau [DP86] found analytically the critical connectivity K_c[1]:

$$\langle Kc \rangle = \frac{1}{2p(1-p)} \tag{3.7}$$

[1]This result is for infinite-sized networks. In practice, for finite-sized networks, the precise criticality point may be slightly shifted.

This can be explained using the simple method of Luque and Solé [LS97]: Focusing on a single node i, the probability that a damage to it will percolate through the network can be calculated. A phase diagram describing this equation can be seen in Fig. 3.3.

It can be seen that this probability will increase with K, as more outputs from a node will increase the chances of damage propagation. Focusing on a node j from the outputs of i, there will be a probability p that $j = 1$. Thus, there will be a probability $1 - p$ that a damage in i will propagate to j. Complementarily, there will be a probability $1 - p$ that $j = 0$, with a probability p that a damage in i will propagate to j. If there are on average K nodes that i can affect, then we can expect damage to spread if $\langle K \rangle 2p(1-p) \leq 1$ [LS97], which implies chaotic dynamics. However, $K = N$ Kauffman Network with $p = 0.5$ has chaotic dynamics, but it is possible to construct RBNs with ordered dynamics, using mechanisms such as canalizing functions[2] [SK04, HSWK02, KPS+].

3.5 Cycles and Attractors

Boolean dynamics correspond to a trajectory within a finite state space of size $\Omega = 2^N$. Any trajectory generated by a dynamical system with immutable dynamical update rules, will eventually lead to a cyclical behavior. No trajectory can generate more than Ω distinct states in a row. Once a state is revisited,

$$\Sigma_t = \Sigma_{t-T}, \qquad T < \Omega,$$

[2] In a canalizing function, the function value is determined when one of its arguments, say $m \in \{1, \ldots, K\}$, is given a specific value, for example, $\sigma_m = 0$. The function value is not specified if the canalizing argument has another value

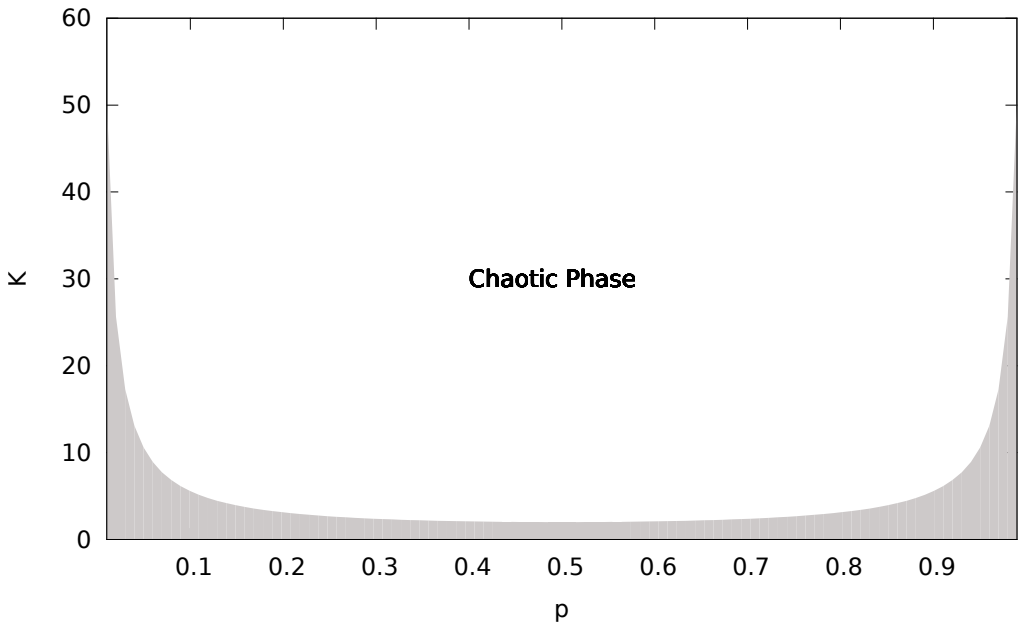


Fig. 3.3: Phase diagram for the classical model, reprinted from [Ald03]

part of the original trajectory is retraced and cyclic behavior follows. The resulting cycle acts as an attractor for an initial conditions set.

Cycles of length 1 are fixpoint attractors. The fixpoint condition $\sigma_i(t+1) = \sigma_i(t) \quad (i = 1, \ldots, N)$ is independent of the updating rules, *viz* synchronous vs. asynchronous. The order of updating the individual σ_i is irrelevant when none of them changes [Ger02].

In Fig. 3.4 a network with $N = 3$ and $K = 2$ is fully defined. The time evolution of the $2^3 = 8$ states of Σ_t is given for synchronous updating. We can observe one cycle of length 2 and two cycles of length 1 (fixpoints).

Attractors. An attractor A_0 of a discrete dynamical system is a region $\{\Sigma_t\} \subset \Omega$ in phase space that maps onto itself under the time evolution $A_{t+1} = A_t \equiv A_0$.

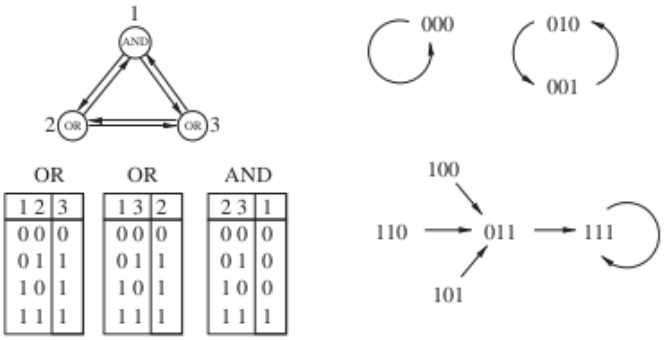


Fig. 3.4: A Boolean network with $N = 3$ sites and connectivities $K_i \equiv 2$. *Left*: Definition of the network linkage and coupling functions. *Right*: The complete network dynamics (from [LS00]).

Attractors are cycles

$$\Sigma^{(1)} \to \Sigma^{(2)} \to \cdots \to \Sigma^{(1)},$$

see Figs.3.5 and 3.4 for some examples. Fixpoints are cycles of length 1.

The Attraction Basin. The attraction basin B of an attractor A_0 is the set $\{\sum_t\} \subset \Omega$ for which there is a time $T < \infty$ such that $\sum_T \in A_0$.

The probability to end up in a given cycle is directly proportional, for randomly drawn initial conditions, to the size of its basin of attraction. The three-site network illustrated in Fig. 3.4 is dominated by the fixpoint $\{1, 1, 1\}$, which is reached with probability 5/8 for random initial starting states.

Attractors and fixpoints are generic features of dynamical systems and are important for their characterization, as they dominate the time evolution in state space within their respective basins of attraction. Random Boolean networks can be used for studies of the dynamics of the system and the structure of network topology.

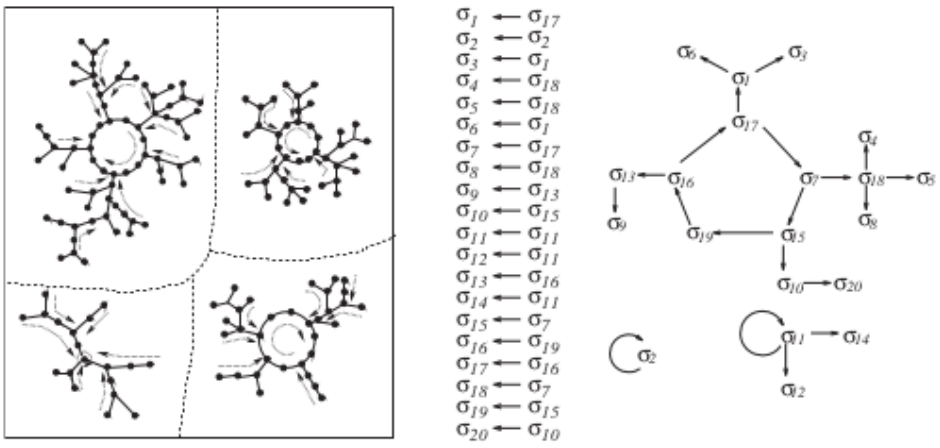


Fig. 3.5: Cycles and linkages. Left: Sketch of the state space where every bold point stands for a state $\Sigma_t = \{\sigma_1, \ldots, \sigma_N\}$. The state space decomposes into distinct attractor basins for each cycle attractor or fixpoint attractor. Right: Linkage loops for an $N = 20$ model with $K = 1$. The controlling elements are listed in the center column. Each arrow points from the controlling element toward the direct descendant. There are three modules of uncoupled variables from [AGCK03].

3.5.1 Linkage Loops, Ancestors and Descendants

Every variable σ_i can appear as an argument in the coupling functions for other elements; in this case it is said to act as a controlling element. The collections of all such linkages can be represented graphically by a directed graph, as illustrated in Figs. 3.4 and 3.5, with the vertices representing the individual binary variables. Any given element σ_i can then influence a large number of different states during the continued time evolution.

Ancestors and Descendants. The elements a variable affects consecutively via the coupling functions are called its descendants. Going backwards in time we find ancestors for each element.

Linkage loops are disjoint for $K = 1$, since they have no element in common, as can be seen in the example of the 20-site network illustrated in Fig. 3.4, where the descendants of σ_{11} are σ_{11} , σ_{12} and σ_{14}.

When an element is its own descendant (and ancestor) it is said to be part of a *linkage loop*. Different linkage loops can overlap, as is the case for the linkage loops

$$\sigma_1 \to \sigma_2 \to \sigma_3 \to \sigma_4 \to \sigma_1, \qquad \sigma_1 \to \sigma_2 \to \sigma_3 \to \sigma_1$$

shown in the example of Fig. 3.6.

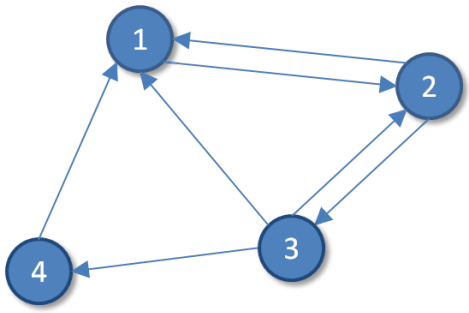


Fig. 3.6: Illustration of a Boolean network with $N = 4$ nodes and a connectivity $K = 2$, and their look up table.

Descendants Elements. The descendants elements set from a Boolean variable σ_i is defined as

$$\varsigma_i = \{\sigma_{i(1)}, \sigma_{i(2)}, \ldots, \sigma_{i(K_{\sigma_i})}\}. \tag{3.8}$$

where K_{σ_i} is the number of Boolean variables that depends of σ_i.

The set of descendants elements might include σ_i itself.

3.5.2 Modules and Time Evolution

The set of ancestors and descendants determines the overall dynamical dependencies.

Module. The collection of all ancestors and descendants of a given element σ_i is called the module (or component) to which σ_i belongs.

If we go through all variables σ_i, $i = 1, \ldots, N$ we find all modules, with every element belonging to one and only one specific module. Otherwise stated, disjoint modules correspond to disjoint subgraphs, the set of all modules constitute the full linkage graph. The time evolution is block-diagonal in terms of modules; $\sigma_i(t)$ is independent of all variables not belonging to its own module, for all times t.

3.5.3 Relevant Nodes and Dynamic Core

Taking a look at dynamics of the 20-site model illustrated in Fig. 3.5, we notice that the elements σ_{12} and σ_{14} just follow the dynamics of σ_{11} , they are *enslaved* by σ_{11} . These two elements do not control any other element and we could just delete them from the system without qualitative changes to the overall dynamics.

Relevant Nodes. A node is termed *relevant* if its state is not constant and if it controls at least one other relevant element (eventually itself).

An element is constant if it always evolves, independently of the initial conditions, to the same state, and it is not constant otherwise. The set of relevant nodes, *the dynamic core*, controls the overall dynamics. The dynamics of all other nodes can be disregarded without changing the attractor structure. The node σ_{13} of the 20-site network illustrated in Fig. 3.5 is relevant if the Boolean function connecting it to itself is either the identity or the negation.

The concept of a dynamic core is of importance for practical applications. Gene expression networks may be composed of thousands of nodes, but contain generally a relatively small dynamic core controlling the overall network dynamics.

3.6 Related Work

Fernández and Solé studied issues related to the evolvability of networks, such as robustness, redundancy, degeneracy, and modularity [FS04]. Furthermore, some studies have shown that topologies affect considerably the properties of RBNs [OS02]. For example, Aldana [Ald03] found that scale-free networks can have critical dynamics with a lower connectivity than expected. A study of modularity was done in [Pob11, PG11], where it was shown that in spite of a high intramodular connectivity, a low connectivity between modules lead the system to the critical regime in what otherwise would be the chaotic regime.

Gershenson et al. [GKS06] studied the effect of redundant nodes on the robustness of RBNs. Using computer simulations, they found that the addition of redundant nodes to RBNs increases their robustness. However, too much redundancy could reduce the adaptability of an evolutionary process. Redundancy is a common feature of engineered systems where redundant parts can substitute for others that malfunction or fail, or augment output when demand for a particular output increases [WJM10].

A previous study about robustness and evolvability in GRNs is presented in [ABKR07]. This work studies the robustness and evolvability of the attractor landscape of GRN, under the process of gene duplication followed by divergence. Robustness is defined as the conservation of attractors under perturbation of the network structure. Evolvability is considered as the emergence of new attractors. The RBN model [Kau69] with *homogeneous random topology* was used, which has been extensively studied, and also a *scale-free output topology*, a more realistic topology [BABC+08]. The results of this work show that an intrinsic property of this kind of networks is that with a high probability,

the previous phenotypes are preserved and new ones may appear. Also, these results indicate that networks operate close to the *critical regime*, exhibiting the maximum robustness and evolvability simultaneously.

Chapter 4

Degeneracy in Random Boolean Networks

Living systems can often maintain functions robustly under volatile conditions, yet also retain a propensity to discover new functions and adapt their functions under novel environments. In other words, an adequate robustness and evolvability is a desirable property in living systems [Gerss].

There are many definitions of degeneracy, but we rely on the definition used in biology, where degeneracy refers to conditions where the functions or capabilities of components overlap partially. As defined in [TSE99], "*degeneracy is the ability of elements that are structurally different to perform the same function or yield the same output; it is a well known characteristic of the genetic code and immune systems*".

There are many examples cited by Edelman and Gally in [EG01], demonstrating the ubiquity of degeneracy in biology. It is pervasive in proteins of every functional class (e.g. enzymatic, structural, or regulatory) [Wag00] and is readily observed in ontogenesis [New94] and the nervous system [TSE99]. Other examples are cited in [Whi10], highlighting the adhesins gene family in saccharomyces process, which expresses proteins that typically play unique roles during development, yet can perform each other's functions when expression levels are altered [GSFF00a]. Another example of degeneracy is found in glucose metabolism, which take place through glycolysis and the pentose

phosphate pathway. These pathways can substitute for each other if necessary even though the sum of their metabolic effects is not identical [SCH+04].

Degeneracy is a ubiquitous biological property that it is a feature of complexity at genetic, cellular, and population levels. Furthermore, degeneracy is a prerequisite of natural selection because natural selection can only operate among a population of genetically dissimilar organisms. This implies that multiple genes contribute in an overlapping fashion to the construction of each phenotypic feature undergoing selection [EG01].

But even if simple redundancy seems to provide robustness versus noise and mutation, from the perspective of evolution, redundancy would make organisms much less able to innovate, because all the copies of components that protect redundant systems probably have to be changed if a change in function is needed, inhibiting the adaptation process. In fact, degeneracy can provide a source of robustness without the drawbacks of redundancy. Thus, the amount of degeneracy can be tuned by evolution to a suitable degree by making the appropriate changes to the network [FS04].

Random Boolean networks, being general models, are used for the analysis of the phase transition in their resulting dynamical space. They are also recognized to be the starting points for the modeling of gene expression and protein regulation networks; the fundamental networks at the basis of all life. Therefore, the study of degeneracy in random Boolean networks, being an intrinsic property of CAS, is the object of study of this work, to understand the dynamics that the RBNs have with nodes that present degeneracy. Important studies of degeneracy effects in complex systems have been made in [TSE99, Whi10, WB10b, WJM10].

This definition of degeneracy can be seen as a *partial redundancy* of the method proposed in [GKS06], as described in the next section. For partial redundancy we refer to a node duplicated of the network, with structural and functional changes.

We define three types of degeneracy from redundant nodes; one of them modifies the function of the network, named *function degeneracy*, while the other two types modify the structure, named *input degeneracy* and *output degeneracy* as described below. It should be noted that a redundant node can be modified by the three types of degeneracy, with a different degeneracy coefficient for each type.

In the section 4.1, we explain how to add a redundant node to a random Boolean network ensemble. Subsequently, we describe the method we proposed to get a degenerated node from the duplicated node.

4.1 Redundancy in RBNs

The process for adding a redundant node to a RBN is the following [GKS06]:

1. Select randomly a node X to be *duplicated*.

2. Add a new node R to the network ($N^{new} = N^{old} + 1$), with the same inputs and lookup table as X (i.e. $k_R = k_X$, $f_R = f_X$), and outputs to the same nodes of which X is input:

$$k_i^{new} = k_i^{old} \cup k_{iR} \text{ if } \exists k_{iX}, \forall i \tag{4.1}$$

3. Double the lookup tables of the nodes of which X is input with the following criterion: When $R = 0$, copy the old lookup table. When $R = 1$, and $X = 0$, copy

the same values for all combinations when $X = 1$ and $R = 0$. Copy again the same values to the combinations where $X = 1$ and $R = 1$. In other words, make an inclusive OR (symbol $\vee$) function in which $X \vee R$ should be one to obtain the old outputs when only X was one.

$$\textbf{if } \exists K_{i_X}, \forall i \begin{cases} f_i^{new}(\sigma_{i_1}, \ldots, \sigma_{i_X} = 0, \sigma_{i_R} = 0 \ldots, \sigma_{i_{K_i}}) = f_i^{old}(\sigma_{i_1}, \ldots, \sigma_{i_X} = 0, \ldots, \sigma_{i_{k_i}}) \\ f_i^{new}(\sigma_{i_1}, \ldots, \sigma_{i_X} = 0, \sigma_{i_R} = 1 \ldots, \sigma_{i_{K_i}}) = f_i^{old}(\sigma_{i_1}, \ldots, \sigma_{i_X} = 1, \ldots, \sigma_{i_{k_i}}) \\ f_i^{new}(\sigma_{i_1}, \ldots, \sigma_{i_X} = 1, \sigma_{i_R} = 0 \ldots, \sigma_{i_{K_i}}) = f_i^{old}(\sigma_{i_1}, \ldots, \sigma_{i_X} = 1, \ldots, \sigma_{i_{k_i}}) \\ f_i^{new}(\sigma_{i_1}, \ldots, \sigma_{i_X} = 1, \sigma_{i_R} = 1 \ldots, \sigma_{i_{K_i}}) = f_i^{old}(\sigma_{i_1}, \ldots, \sigma_{i_X} = 1, \ldots, \sigma_{i_{k_i}}) \end{cases}$$

After this process, R will be a redundant node of X, and vice versa. A diagram illustrating the inclusion of a red node is depicted in Fig. 4.1.

Example of Degeneracy in a RBN

To illustrate the three types of degeneracy, we use an example. Let us define a RBN with $N = 3$ and $K = 2$. The *Controlling Elements* for each one are given by

$$X = f(Z, Y),$$

$$Y = f(W, X),$$

and

$$Z = f(X, Z)$$

respectively. Adding a redundant node, the new functionalities for Y and Z are defined as

$$Y^r = f(W, X \vee R),$$

and

$$Z^r = f(X \vee R, Z)$$

respectively, while Controlling Elements σ_X are the same. For the added node R, the Controlling Elements are given by the function $R = f(Z, Y)$ as seen in 4.1.

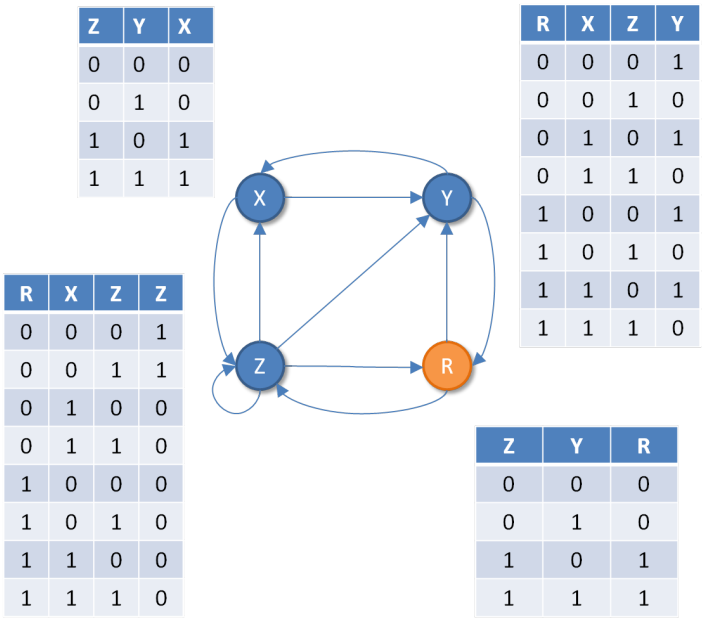


Fig. 4.1: A simple RBN with $N = 4$, $K = 2$, with $N_{red} = 1$.

4.2 Degeneracy in RBNs

After adding a redundant node to the RBN ensemble, we proceed to add degeneracy to the duplicated node. We require three parameters which are the coefficients for:

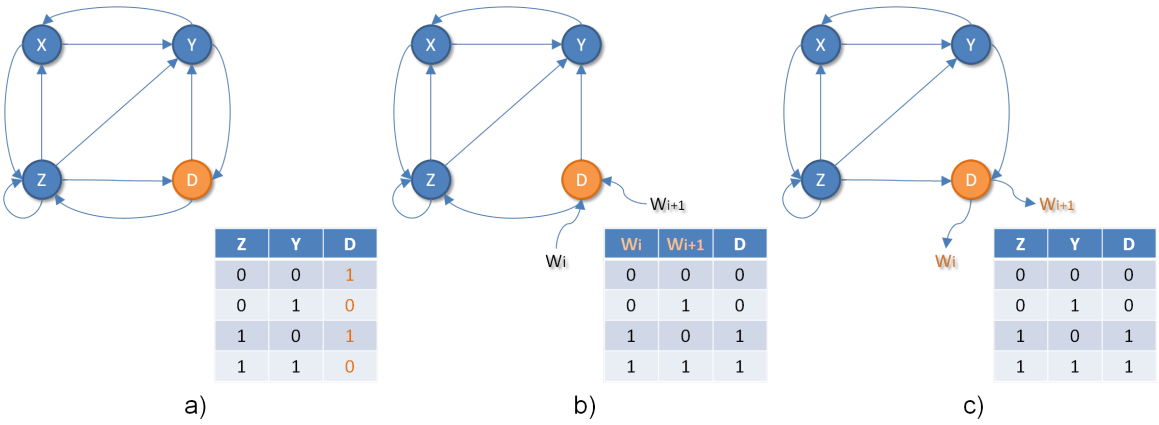
$\phi \in (0, 1)$ degeneracy in *coupling functions*. This variable represents the coefficient of change in the coupling function,

$\iota \in (0, 1)$ degeneracy in *connecting input links*. This variable represents the coefficient of change of input controlling elements.

ω degeneracy in *connecting output links*. This variable represents the coefficient of change of output controlling elements, with respect to the duplicated node.

In this way, the new network has $N+1$ Boolean variables, considering the RBN ensemble plus the new redundant node added, maintaining the same connectivity K. Such new redundant node will keep a similarity with respect to the selected node, changing some values as function of ϕ in the case of *coupling function*, ω for changes in the *connecting input links* and ι for changes in the *connecting output links*.

Adding a node to a RBN, if we have $\phi = \iota = \omega = 0$, it means that the new node differs from the input and output links with respect to the selected node, as well as it lookup table (the new values generated are random, so some of these values may match), as seen in figure 4.2. On the other hand, if we have $\phi = \iota = \omega = 1$ it means that the input, output, and the lookup table is equivalent to the selected node, yielding a redundant node as in figure 4.1. In other words, the value of the variable is analogous to the degree of correlation between the selected node, and the new node.



Z	Y	D
0	0	1
0	1	0
1	0	1
1	1	0

Wi	Wi+1	D
0	0	0
0	1	0
1	0	1
1	1	1

Z	Y	D
0	0	0
0	1	0
1	0	1
1	1	1

Fig. 4.2: RBN of $N = 4$. Node D has a) function, b) input and c) output degeneracy, respectively. Degeneracy in D is from adding a redundant node of X.

When we add a node, for some σ_i, the number of controlling elements increases from K_i to $K_i + 1$. Thus, the number of states for the controlling function is of $2^{K+1} = 2^{k'}$. In the case that a node gets a controlling element from $\sigma_{i_{red}}$, its coupling function is doubled, as a canalizing function.

In the following sections, we present the definition of redundancy used in this work, and our proposed definition of degeneracy. An example of degeneracy can be seen in fig. 4.1.

4.3 Function Degeneracy

To define the function degeneracy, firstly let's define the number of ones in a coupling function as:

$$l = \sum_{i=1}^{N} \sigma_i \tag{4.2}$$

We define $\phi \in [0, 1]$ as the coefficient of degeneracy in a coupling function. A randomly selected node is duplicated, maintaining the same input and output links, but changing $z_\phi = \lfloor l \cdot (1 - \phi) \rfloor$ number of variables to the new coupling function $f_{i_{fd}}$, which are randomly selected. The new values to assign are randomly selected with probability p.

Therefore, if we have $\phi = 1$, then $f_i = f_{i_{fd}}$. Otherwise, if $\phi = 0$, then all the values (2^{K_i} variables) of $f_{i_{fd}}$ will be assigned randomly.

For the new value to be assigned, let's define

$$d : \{0, 1\}^* \to [0, \infty) \tag{4.3}$$

such that d_i can be either 0 or 1 with probability $p(1) = p(0) = 0.5$.

Using the Eq. 3.3 and Eq. 4.3, the coupling function of $f_{i_{fd}}$ is

$$f_{i_{fd}} = \{\sigma_{j_1(i)}, \quad \sigma_{j_2(i)}, \quad \ldots, \\ \sigma_{j_{m_1}(i)} = d_1, \sigma_{j_{m_2}(i)} = d_2, \ldots, \sigma_{j_{m_{z_\phi}}(i)} = d_{Z_\phi}), \\ \ldots, \quad \sigma_{j_K(i)}\} \rightarrow \sigma_{i_{fd}}$$

Note that the values of m are randomly selected, having $1 \leq m \leq 2^{K_i}$.

Using the example of Fig. 4.1, the Fig. 4.2 a) shows a RBN with function degeneracy in the duplicated node.

4.4 Input Degeneracy

This kind of degeneracy modifies the structure. For input degeneracy, a randomly selected node is duplicated, maintaining the same function and outputs links, but changing the $z_\iota = \lfloor \dot{K(1-\iota)} \rfloor$ number of input connecting links (also called *controlling elements*). For this case, $\iota \in [0, 1]$ is the coefficient of degeneracy in the controlling elements (see eq. 3.2) of a Boolean variable σ_i.

Similar to function degeneracy, if we have $\iota = 1$, then $\sigma_i = \sigma_{i_{id}}$. Otherwise, if $\iota = 0$, then all the controlling elements $\sigma_{j_1(i)}, \sigma_{j_1(i)}, \ldots, \sigma_{j_{K_i(i)}}$ will be randomly re-assigned.

Using the Eq. 3.2 and Eq. 4.3, the time evolution of a Boolean variable $\sigma_{i_{id}}$ is given by

$$\sigma_{i_{id}} = f_i(\sigma_{j_1(i)}, \sigma_{j_2(i)}, \ldots, \sigma_{j_{m_1}(i)}, \sigma_{j_{m_2}(i)}, \ldots, \sigma_{j_{m_{z_\iota}}(i)}, \ldots, \sigma_{j_K(i)})$$

Note that the values of m are randomly selected, having $0 \leq m \leq K_i$.

Using the example of fig 4.1, the fig. 4.2 b) shows a RBN with function degeneracy in the duplicated node.

4.5 Output Degeneracy

Output Degeneracy, as input degeneracy, modifies the structure of an RBN. In this case, a randomly selected node is duplicated, maintaining the same *coupling function* and *controlling elements*, but changing $z_\omega = \lfloor K_{out}\dot{(}1 - \iota)\rfloor$ output connection links number of outputs links. For this case, $\omega \in [0, 1]$ is the coefficient of degeneracy in the output links.

As with previous description of partial redundancy, if we have $\omega = 1$, then $\sigma_i = \sigma_{i_{od}}$. Otherwise, if $\iota = 0$, then all the descendant controlling elements $\sigma_{i(1)}, \sigma_{i(1)}, \ldots, \sigma_{i(K_i)}$ will be re-assigned randomly.

Using the Eq. 3.2 and Eq. 4.3, the direct descendant set of output controlling elements $\varsigma_{i_{od}}$ is determined by

$$\varsigma_{i_{od}} = \{\sigma_{i(1)}, \sigma_{i(2)}, \ldots, \sigma_{i(m_1)}, \sigma_{i(m_2)}, \ldots, \sigma_{i(m_{z_\omega})}, \ldots, \sigma_{i(K_{\sigma_i})}\}$$

Note that the values of m are randomly selected, having $0 \leq m \leq K_{\sigma_i}$.

Chapter 5

Numerical Simulations

As described in previous chapters, the behavior of artificial and biological networks is determined by the dynamics of its attractor states. This behavior, is related to the notions of order, complexity and chaos, which depend largely on the degree to which their attractors converge [Wue02].

This chapter exhibits results obtained to measure number of attractors, size of attractors, sensitivity to initial conditions and finally, measures of information theory and statistics for classic RBN model, adding redundant nodes with several values for the coefficients of degeneracy, as defined in Chap. 4. The open software laboratory RBNLab [Ger05] was extended to implement and thus explore the properties of degeneracy mentioned before. The improved version of RBNLab and its Java source code are available at `http://rbn.sourceforge.net`.

In this chapter, we call RBN core, an ensemble of a classical RBN [Kau69] with size N, K connectivity, uniform distribution and a homogeneous topology. For all experiments, the probability for getting a 1 in the rules is $p = 0.5$ (no bias in lookup tables [Ger04a]). The experiments we performed start by creating RBN core ensembles, adding redundant nodes to them, with different values for ι, ϕ, ω coefficients, to improve degeneracy, according to the definition in Chap. 4. We refer to N_{deg} to the number of nodes added with certain degree of degeneracy.

To calculate statistical properties, we randomly generated 1000 networks. RBN core ensembles with $N = 15$ and $N_{deg} = 5$, with $K = \{1, 2, 3, 4, 5\}$. We show the figures for the results with $K = 5$ because these are the most illustrative, and in the appendix A the summary of numerical results is shown. The results compare with redundant nodes and simple RBN ensembles. It should be noted that results presented are based on the dynamics of the RBN core ensembles, i.e. results do not consider the controlling elements of N_{deg}.

Experiments with ι, ϕ, ω values between $(0, 1)$ were performed, but the effects of degeneracy showed in the results are not statistically significant and not illustrative, being differences between treatments less pronounced.

5.1 Statistical properties

The statistical properties that were considered to determine the dynamics of RBN ensembles with partial redundant nodes are the following:

- Number of attractors (A).
- Attractors average length ($\bar{L}$).
- Percentage of states in attractors ($\%SIA$).

A description of the experiments and their results are described below. For practical reasons, we will call in graphs *Simple* to RBN ensembles without any kind of degeneracy, *FD* to RBN ensembles with function degeneracy, *ID* to RBN ensembles with Input

Degeneracy, *OD* to RBN ensembles with output degeneracy and *Red* to RBN ensembles with Redundant nodes.

5.1.1 Number of Attractors (A).

There is evidence that cell types correspond to different dynamical states of a complex system, i.e. the gene expression network. The number of attractors reflects how many distinct sets of states can *capture* the dynamics of the RBN. When $A > 1$ it is considered that the system is multistable [Tho78]. It can be said that more attractors imply more potential functionality of the network.

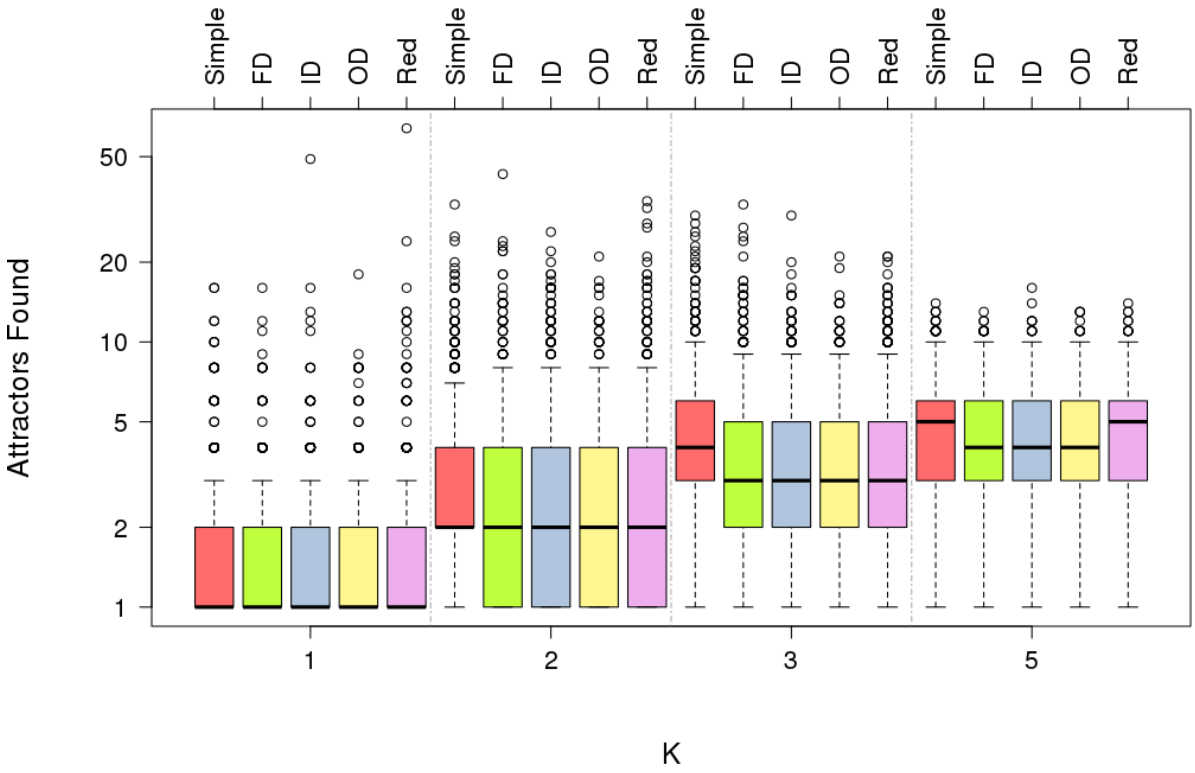


Fig. 5.1: Number of Attractors Found with $N = 15$, $N_{deg} = 5$, $K = \{1, 2, 3, 5\}$; in logarithmic scale.

In Fig. 5.1 we can see the results for RBN ensembles with $N = 15$, $N_{deg} = 5$ and $K = 5$

and a larger RBN with $N = 80$, $N_{deg} = 20$ and $K = 3$, showing the number of attractors found, presented as boxplots[1].

These results shows that any type of degeneracy in RBNs affects the number of attractors, while redundancy maintains the same number of attractors as the RBNs core. It should be noted that statistical results give overlapping variances for $N = 15$ cases.

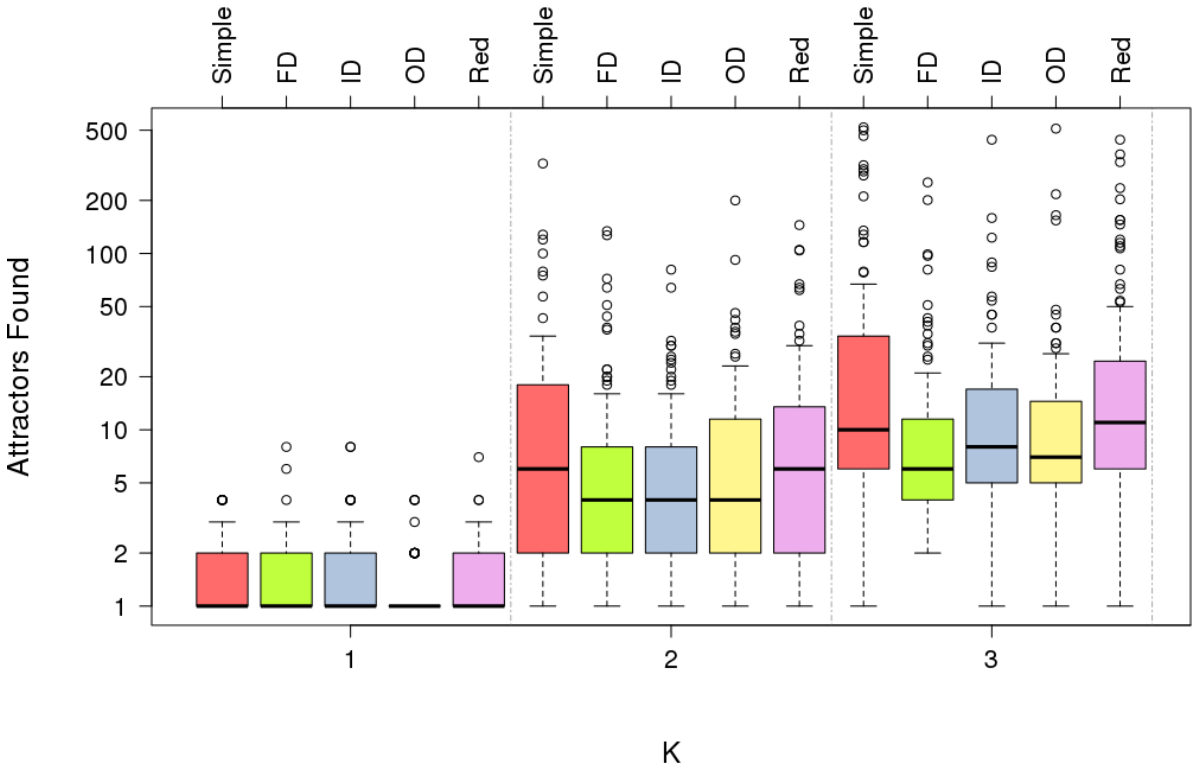


Fig. 5.2: Number of Attractors Found with $N = 80$, $N_{deg} = 20$, $K = \{1, 2, 3\}$; in logarithmic scale.

The effect of degeneracy is stronger on RBNs with a high connectivity K, i.e. in the

[1]A boxplot is a non-parametric representation of a statistical distribution. Each box contains the following information: The median ($Q2 = x_{0.50}$) is represented by the horizontal line inside the box. The lower edge of the box represents the lower quartile ($Q1 = x_{0.25}$) and the upper edge represents the upper quartile ($Q3 = x_{0.75}$). The interquartile range ($IQR = x_{0.75} - x_{0.25}$) is represented by the height of the box. Data which is less than $Q1 - 1.5 \cdot IQR$ or greater than $Q3 + 1.5 \cdot IQR$ is considered an *outlier*, and is indicated with circles. The *whiskers* (horizontal lines connected to the box) show the smallest and largest values that are not outliers.

chaotic regime. RBNs in the ordered and critical regimes ($K \leq 2$) seem not to be affected by the addition of any type of degenerate nodes, as can be seen in Fig. 5.1. Results for larger networks can be seen in Fig. 5.2 (For more details of results, see summary tables in Appendix A).

5.1.2 Average Attractors Lengths $\bar{L}$

When $L = 1$, there are only point attractors in the network. Larger values of L indicate longer cycle attractors.

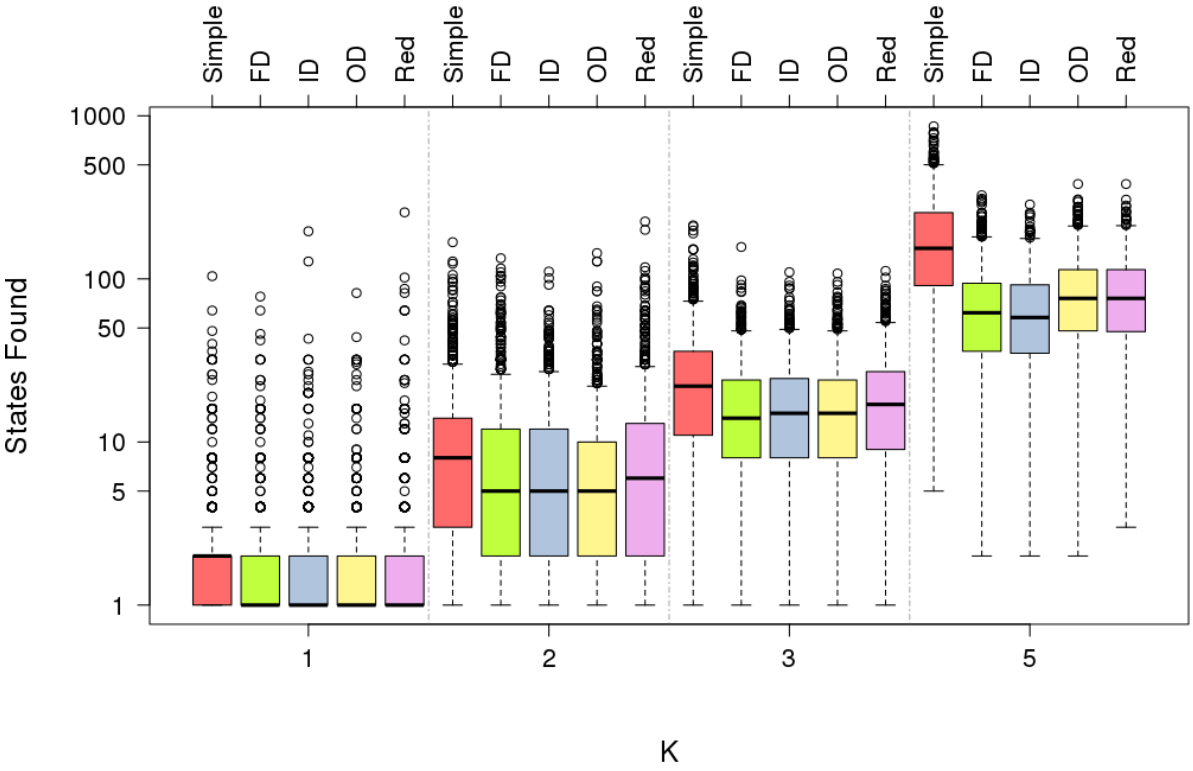


Fig. 5.3: $\bar{L}$ for RBN ensembles with $N = 15$, $N_{deg} = 5$, $K = \{1, 2, 3, 5\}$; in logarithmic scale.

In general, results obtained of RBNs with degenerated nodes, have attractors with smaller number of states. Moreover, the effect of degeneracy in the RBN seems to be less in networks with redundant nodes and output degeneracy, and higher for networks

with nodes with function and input degeneracy, as shown in the Fig. 5.3. The results are more significant for larger networks, although the high standard deviation should be noted in all cases.

This can be explained by the fact that in deterministic updating schemes (as the one used in this work), attractor lengths grow algebraically with N [Ger04b]. Larger networks allow the possibility of more combinations of states in an attractor, increasing its length. It is not so much that a large N favors longer attractors, but a small N restricts their possibility. For $K \leq 2$, classical RBNs have the lowest L. Results for simple RBNs and ensembles with redundant nodes have longer attractors, in larger networks with $N = 80$, $N_{deg} = 20$ and $K = 2$, as can be seen in Fig. 5.4. More results can be seen in the Appendix. These results with larger networks ($N = 720$) are less significative because they represent the results for 100 RBN ensembles.

5.1.3 Average States in Attractors $\%SIA$.

The Percentage of States in Attractors ($\%SIA$) reflects how much *complexity reduction* is performed by the network, i.e. from all possible $\Omega = 2^N$ states, how many states *capture* the dynamics. It depends on the number of attractors and their length. Even when complexity reduction is relevant, larger values of $\%SIA$ indicate a more complex potential functionality of the network, i.e. richer dynamics [PG11]. A large $\%SIA$ can be given by large attractor lengths $\bar{L}$ and/or a high number of attractors A. The more and longer attractors a network has, it can exhibit a richer behavior.

Redundancy decreases the $\%SIA$, in the chaotic regime, as it can be seen for $K = 5$ in Fig. 5.5. Degeneracy reduce $\%SIA$, even more: output degeneracy decreases the most,

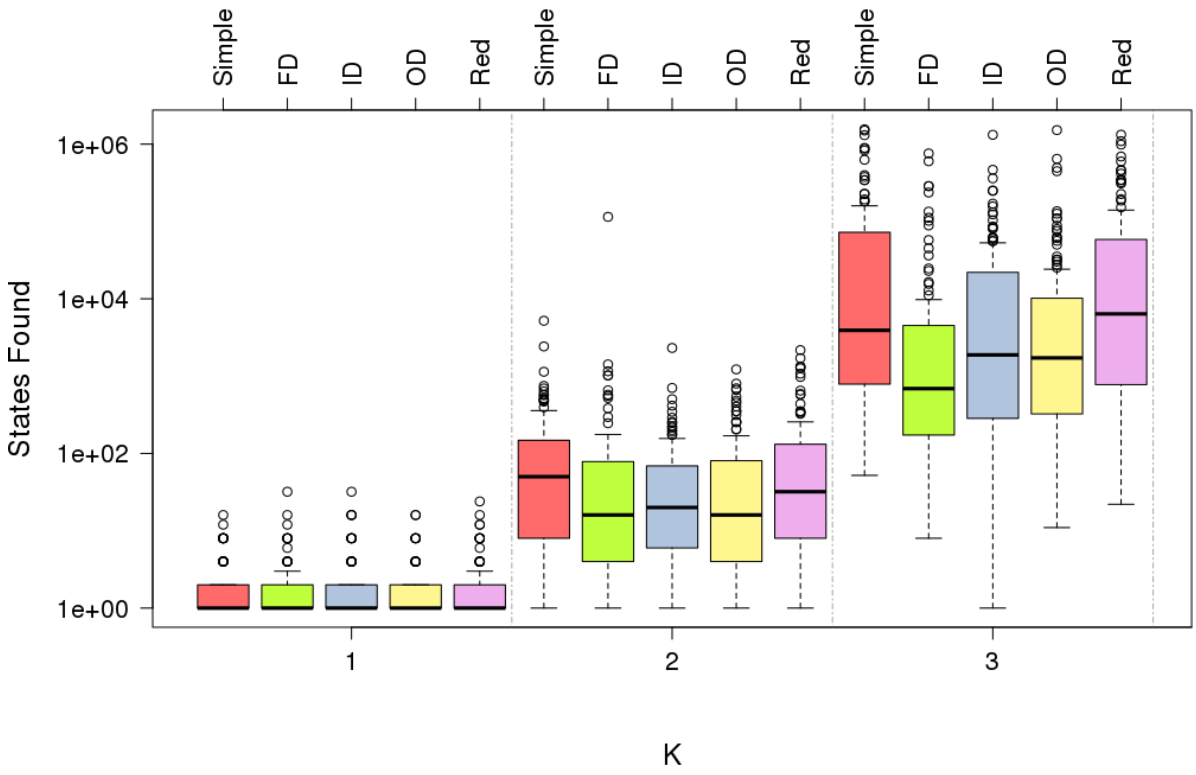


Fig. 5.4: $\bar{L}$ for RBN ensembles with $N = 80$, $N_{deg} = 20$, $K = \{1, 2, 3\}$; in logarithmic scale.

while function degeneracy decreases the least. Since the number of attractors remained almost constant, we can deduce that a reduction of SIA is due to a reduction of the attractor lengths.

$$\%SIA = 100 \cdot \frac{A \cdot L_e}{2^N}. \tag{5.1}$$

Statistical results often give high standard deviations σ. This is because some networks might have a single point attractor, while others might have several cycle attractors. However, even the averaged values are informative: they show the effect of RBNs with different parameters on the network properties and dynamics. Nevertheless, the potential implications of high standard deviations should not be discarded.

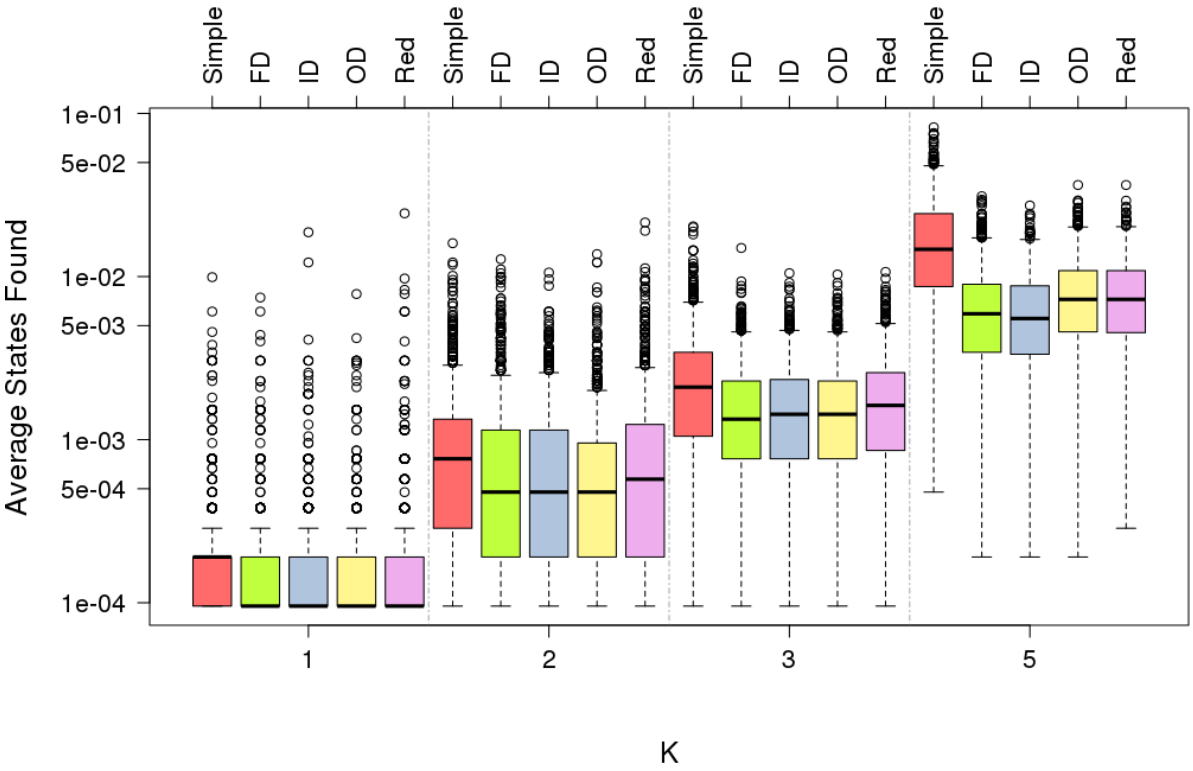


Fig. 5.5: Percentage of states in Attractors (%SIA) for RBN ensembles with $N = 15$, $N_{deg} = 5$ $K = \{1, 2, 3, 5\}$; in logarithmic scale.

5.2 Sensitivity to Initial Conditions

To characterize the dynamical regime of an ensemble of discrete dynamical systems, in this case RBNs, we measured the sensitivity to initial conditions; measuring how small differences in initial states lead to similar or different states: if the system is sensitive to small differences, it is considered to be in the chaotic regime. This method is similar to stability analysis of dynamical systems [Sey94] or damage spreading [Sta94]. There are also equivalents to Lyapunov exponents in RBNs [LS00]. The rationale is similar in all of these methods: if perturbations do not propagate, then the system is in the ordered dynamical phase. If perturbations propagate through the system, then it is in the chaotic dynamical phase [PG11]. The phase transition (also called critical regime) lies where the size of the perturbation remains constant in time.

To measure the effects of spreading damage in RBNs with degeneracy, we made experiments of sensitivity to initial conditions, with RBN ensembles of size $N = 20$, having $N_{deg} = 5$ nodes with degeneracy, for different values. The number of states analyzed was $2^{15} = 32768$, which is the number of total states in these network ensembles.

The process we followed to measure statistically the sensitivity to initial conditions of RBNs, is described in the next steps [Ger04b]:

1. For a RBN ensemble with $N = 15$ and a random initial state σ_i, let it run for a number of steps t_{max} (for our experiments, $t_{max} = 10,000$), to reach a final state $\Sigma'(t_{max}) = \sigma'$.

2. After that, we apply a random point mutation to initial state σ_i to obtain σ'_i , i.e. flipping a bit of σ_i.

3. Then, let the network run for t_{max} from σ'_i and σ_i to obtain the final states σ_f, σ'_f, respectively.

The difference between states σ_i and σ_f, and between σ'_i and σ'_f were calculated using the normalized Hamming distance, which measures the number of units that are different in two set of states:

$$H(\sigma, \sigma') = \frac{1}{N} \sum_{i=1}^{N} |\sigma_i - \sigma'_i|. \tag{5.2}$$

If states σ and σ' are equal, then $H(\sigma, \sigma') = 0$. The maximum $H = 1$ is given when σ is the complementary state of σ', i.e. full anticorrelation. $H = 0.5$ implies no correlation between σ and σ'. The smaller H is, the more similar σ and σ' are. As H increases (up to $H = 0.5$), it implies that differences between σ and σ' also increase.

Because there is only one bit difference between σ_i and σ'_i and each state has N bits:

$$H_i = H(\sigma_i, \sigma'_i) = \frac{1}{N}. \tag{5.3}$$

Now, to measure the sensitivity to initial conditions, the difference between the final and initial Hamming distances ΔH is used:

$$\Delta H = H_f - H_i, \tag{5.4}$$

where

$$H_f = H(\sigma_f, \sigma'_f). \tag{5.5}$$

When $\Delta H < 0$, it means that different initial states converge to the same final state, a characteristic of the ordered regime, where trajectories in state space tend to converge. when $\Delta H > 0$, it means that small differences in initial states tend to increase, a characteristic of the chaotic regime, where trajectories in state space tend to diverge. When $\Delta H = 0$, small initial differences are maintained, a characteristic of the critical regime, where trajectories in state space neither converge nor diverge (in practice, $\Delta H \approx 0$). Thus, the average ΔH can indicate the regime of a RBN. Boxplots of results are shown in Fig. 5.6. Note that boxplots show medians. Statistics results can be found in Appendix A.

For $K = 1$, the dynamics are in the ordered regime for all cases, since small differences in initial states tend to be reduced, indicated by a negative $\Delta H = -\frac{1}{N_T}$. For $K = 2$, the average ΔH is close to zero for all cases, suggesting a critical regime. The difference caused by degeneracy is clearly seen for $K > 2$, i.e. in the chaotic regime. Fig. 5.7 shows the average of ΔH for RBN ensembles with $N = 20$, $N_{deg} = 5$ and $K = \{1, 2, 3, 5\}$

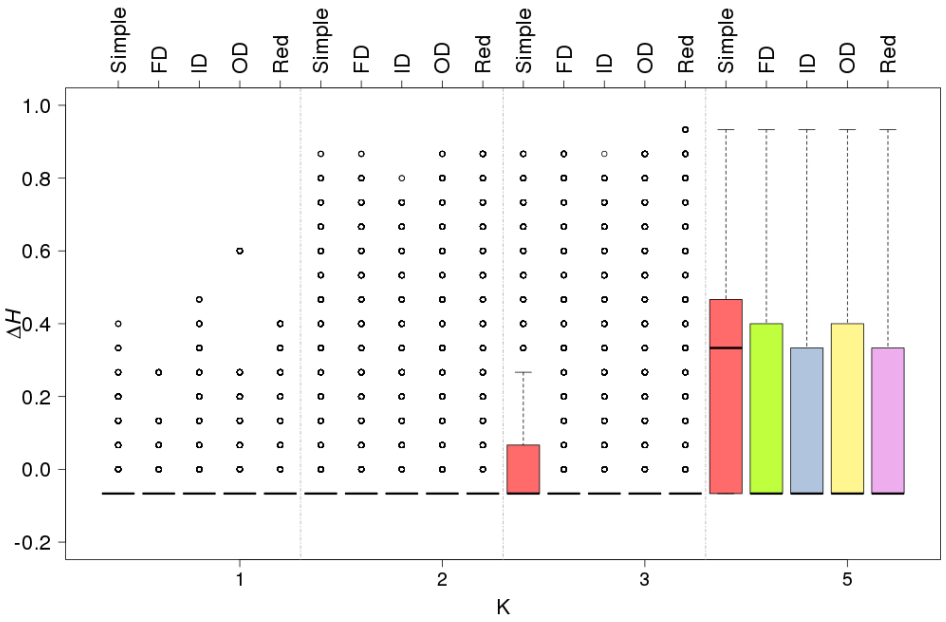
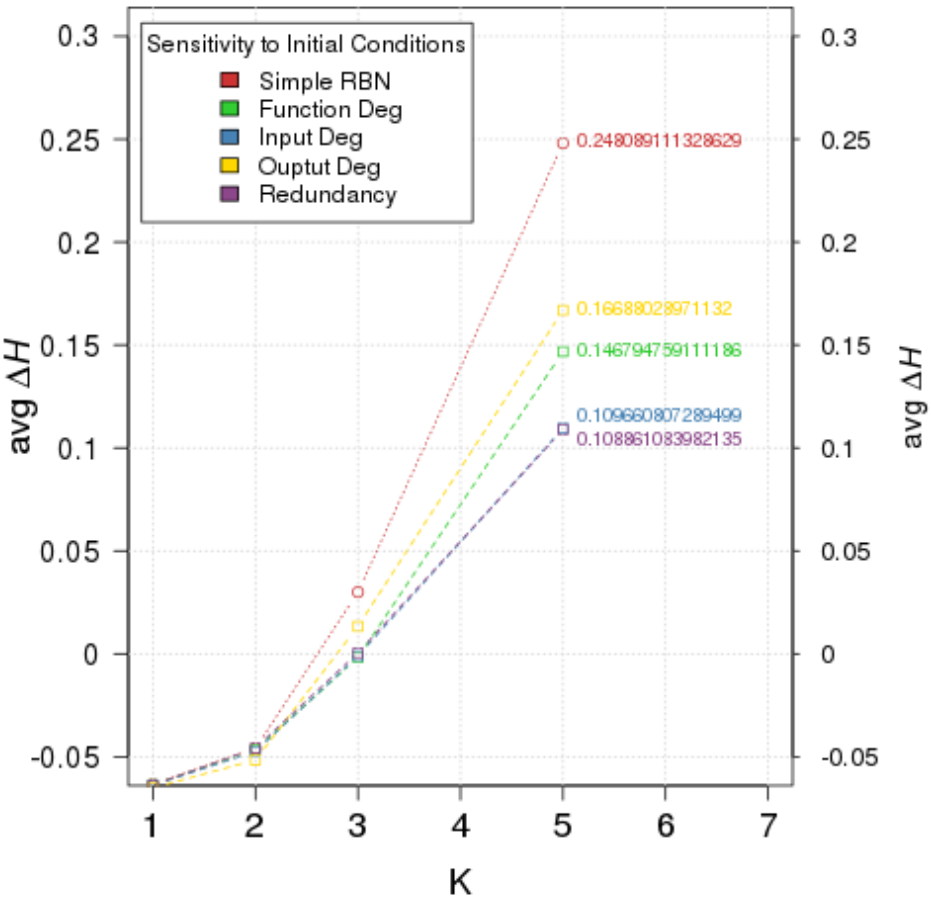


Fig. 5.6: Average of Sensitivity to Initial conditions for $N = 20$, $N_{deg} = 5$, $K = \{1, 2, 3, 5\}$; for 1000 RBN ensembles.

The sensitivity to initial conditions is related to the robustness provided by redundancy and partial redundancy, in cases of function degeneracy, input degeneracy, with values close to zero. This behavior is because it is more difficult for damage to spread through the network, having nodes which provide stability to the descendant nodes. In the case of output degeneracy, damage spreading is potentially greater in the sense that the states affect greater number of descendant nodes.



x

Fig. 5.7: Average of Sensitivity to Initial conditions for $N = 20$, $N_{deg} = 5$, $K = \{1, 2, 3, 5\}$; for 100 RBN ensembles.

To ensure that the results with $N = 20$ were not an artifact of the small size of the networks, we also executed experiments for larger networks with $N = 80$, which 20% of its nodes have partial redundancy, analyzing one hundred networks and one hundred state pairs which were explored for ten thousand steps. These experiments are less statistically significant, but they clearly show that degeneracy and redundancy provides a significant difference in sensitivity to initial conditions, and not to network size. Results can be observed in Fig. 5.8 and Fig. 5.9.

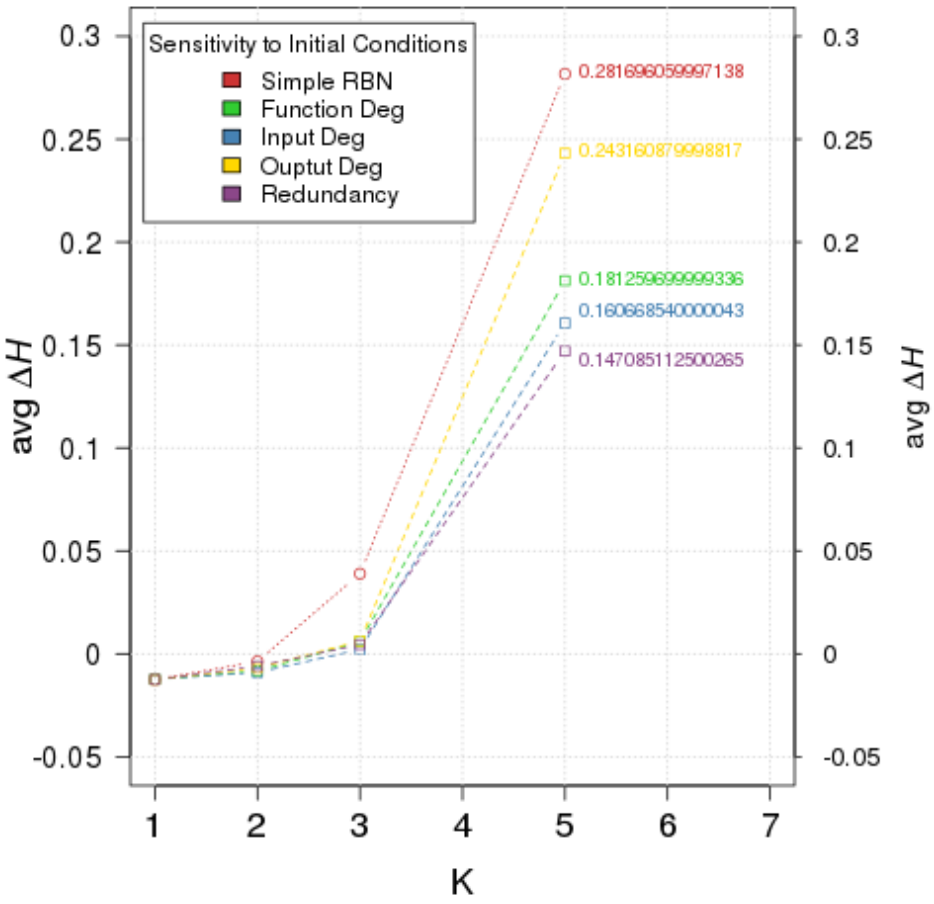


Fig. 5.8: Average of sensitivity to initial conditions for $N = 100$, $K = \{1, 2, 3, 5\}$, $N_{deg} = 20$, for 1000 RBN ensembles.

5.2.1 Adding Degeneracy plus redundancy

The results obtained in previous experiments, have led us to ask the question of whether the effects of redundancy and degeneracy are additive, i.e., what happens if we add redundant nodes plus one kind of degeneracy to a RBN? In this section we show the results of adding redundancy and degeneracy in RBN ensembles. The way in which the nodes were added, were choosing a random node, and doubling it twice. One of these duplicated nodes is the redundant node, and the other duplicated node has one of the three types of degeneracy.

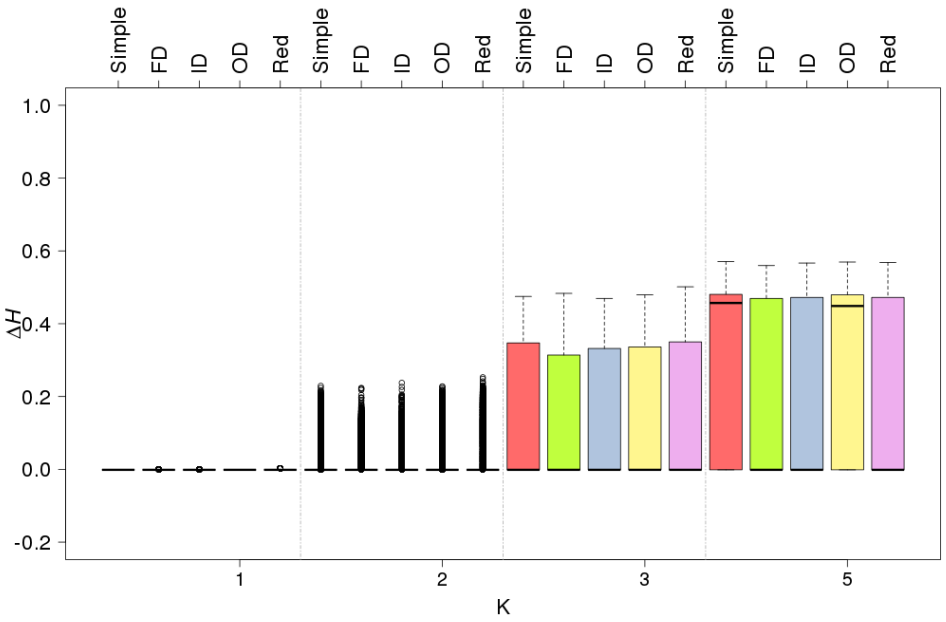


Fig. 5.9: Average of Sensitivity to Initial conditions for $N = 800$, $N_{deg} = 80$, $K = \{1, 2, 3, 5\}$; for 100 RBN ensembles.

In this section, we present the results of sensitivity to initial conditions in RBNs, having $N = 15$ and $K = \{1, 2, 3, 5\}$, adding $N_{red} = 5$ redundant nodes and $N_{deg} = 5$ nodes with degeneracy. These results are compared with RBN ensembles with $N = 15$, called RBN core, as can be seen in Fig. 5.10 and 5.11. One thousand RBN ensembles were performed. The numerical results are shown in appendix A.

We also executed experiments with larger networks, having $N = 880$, $N_{red} = 80$, $N_{deg} = 80$ and $K = \{1, 2, 3, 5\}$, for one hundred RBN ensembles, where we can see a greater δH difference with respect to RBN core ensembles.

The experiments of RBN ensembles with degeneracy plus redundancy, shows a greater effect of having nodes with redundancy plus some kind of degeneracy, than just having

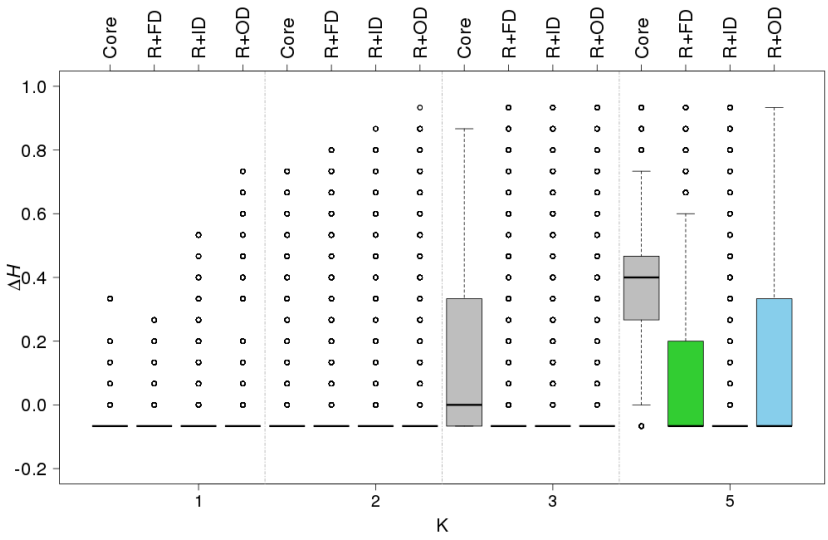


Fig. 5.10: Boxplot of sensitivity to initial conditions for $N = 15$, $K = \{1, 2, 3, 5\}$, $N_{red,deg} = \{5, 5\}$, for 1000 RBN ensembles.

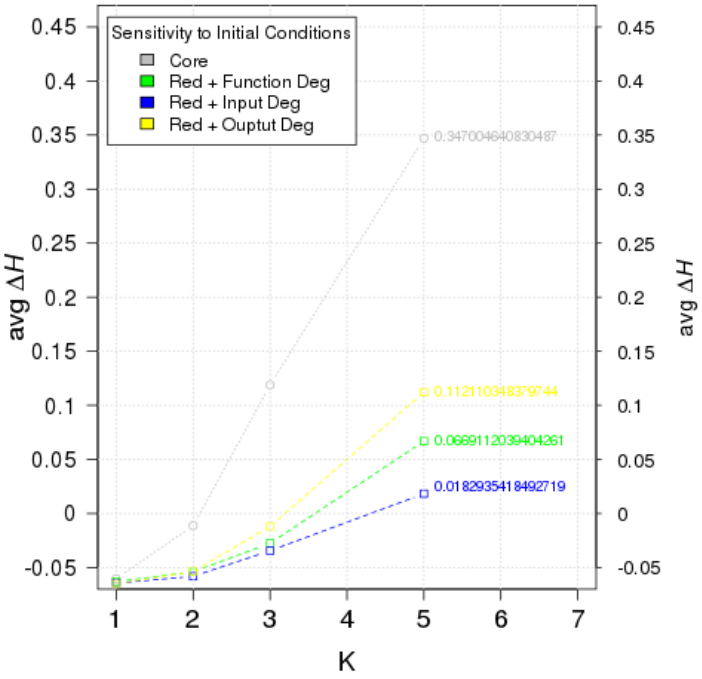


Fig. 5.11: Average of sensitivity to initial conditions for $N = 15$, $K = \{1, 2, 3, 5\}$, $N_{red,deg} = \{5, 5\}$, for 1000 RBN ensembles.

redundancy or degeneracy, but we should consider than the size of the networks is greater, suggesting than the effect of robustness is additive. Also, these results allow to

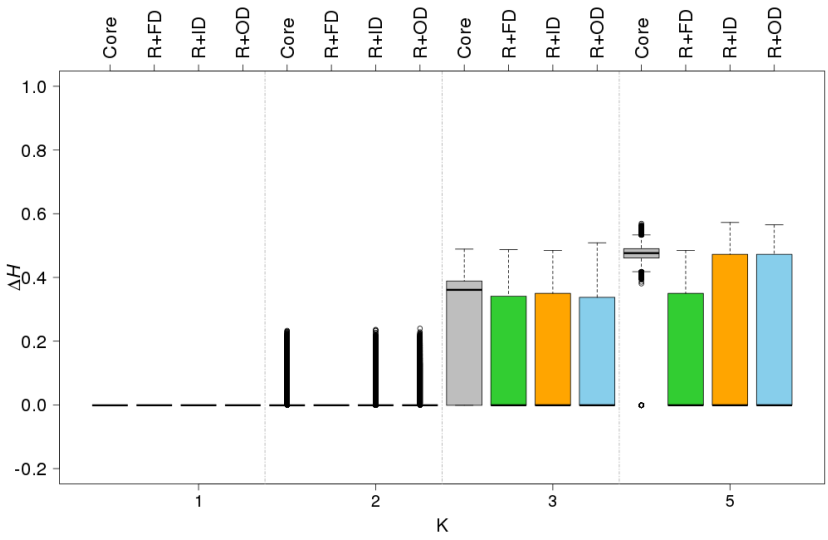


Fig. 5.12: Boxplot of sensitivity to initial conditions for $N = 720$, $K = \{1, 2, 3, 5\}$, $N_{red,deg} = \{5, 5\}$, for 100 RBN ensembles.

observe that RBN ensembles with degeneracy and redundancy provide higher stability to sensitivity to initial conditions than simple RBNs, as shown in Fig. 5.13 and 5.12

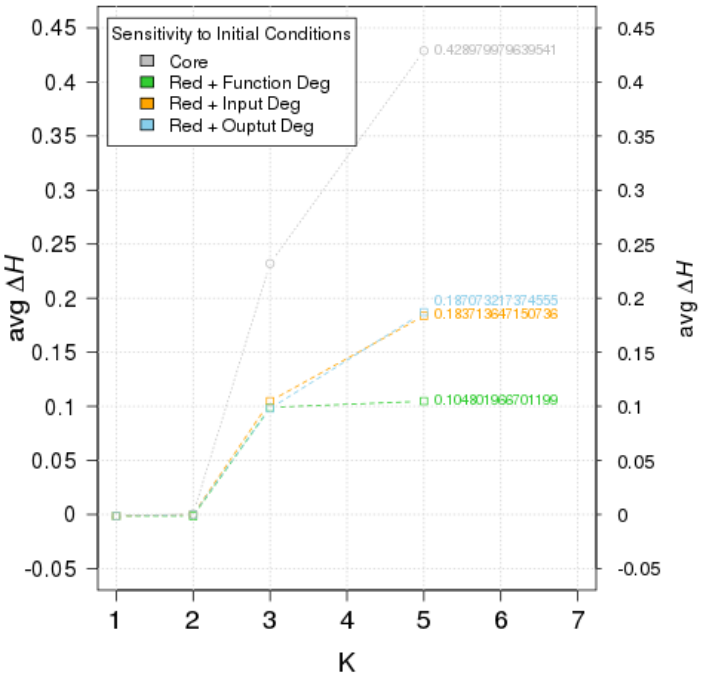


Fig. 5.13: Average of sensitivity to initial conditions for $N = 720$, $K = \{1, 2, 3, 5\}$, $N_{red,deg} = \{5, 5\}$, for 100 RBN ensembles.

5.3 Measures of complexity

Entropy is a concept from physics which encodes the multiplicity of states available to a system at a given temperature. Entropy is also a central concept in information theory, where it is commonly denoted as *Shannon entropy* or *information entropy*. In this context, we are interested in the amount of information encoded by a sequence of symbols of an alphabet, e.g. when transmitting a message:

$$\ldots, \sigma_{t+2}, \sigma_{t+1}, \sigma_t, \sigma_{t-1}, \sigma_{t-2}, \ldots.$$

Let us consider two time series of an alphabet composed by $\{0, 1\}$, e.g.

$$\ldots, 101010101010, \ldots, \qquad \ldots, 1100010101100, \ldots \tag{5.6}$$

In these examples, entropy is the average amount of *surprise* associated with each time series. The amount of *surprise* of a particular event is a function of the probability of that event. The less probable an event, the more surprising it is.

In 1948, Claude Shannon studied in the context of telecommunication. Information can be represented with a string X, composed by a sequence of values x which follow a probability distribution $P(x)$.

$$I = -\sum P(x) log P(x). \tag{5.7}$$

Shannon information can be seen also as a measure of uncertainty. If there is absolute certainty about the future values of x, then, the information received will be zero. If the probability distribution of receiving either zero or one is $P(0) = P(1) = 0.5$, then

the information received is maximal.

From this definition, in [GF12] Emergence E is defined as the amount of information produced by a process.

$$E = \frac{I_{out}}{I_{in}}, \tag{5.8}$$

where I_{in} is the *input information* and I_{out} is the *output information*, which can be seen as $I_{out} = f(I_{in})$.

In the other hand, the measure of self-organization was proposed in [GHA03] as the negative of the change of information δI: if the information is reduced, then self-organization occurs, while an increase of information implies self-disorganization.

$$S = I_{in} - I_{out}. \tag{5.9}$$

From the definition of information, complexity can be seen as the amount of information required to describe a phenomenon at a particular scale [By04]. If more information is required then the complexity can be said to be higher, considering the scale of observation (which is not discussed in this work). This work is based on the definition proposed by [LRMC10], where complexity is the multiplication of Shannon's Information and disequilibrium.

$$E = E * S. \tag{5.10}$$

High E implies a low S and vice versa. For all cases, $I_{in} = 1$, is assumed, due to random inputs.

A final measure made was homeostasis, defined as the ability of an organism to main-

tain steady states of operation, in view of the internal and external changes [Can32]. Homeostasis is also defined as the opposite of a normalized Hamming distance:

$$H = 1 - d(I_{in}, I_{out}) \tag{5.11}$$

The results obtained for the measures, as defined in [GF12] can be seen in Fig. 5.14.

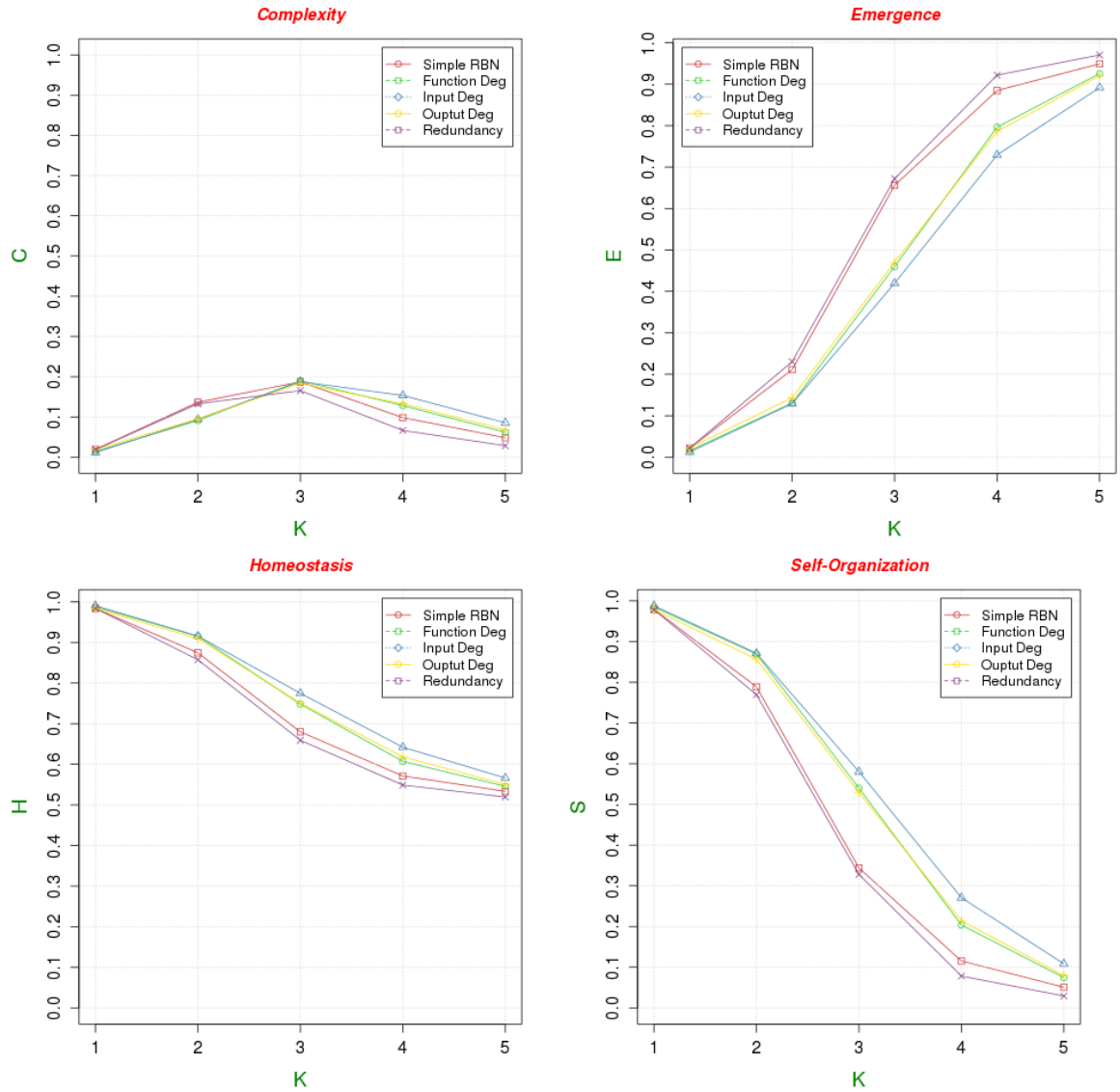


Fig. 5.14: Measures of a) complexity, b) emergence, c) self-organization, and d) homeostasis. RBN ensembles with $N = 20$ with $N_{deg} = 5$, $K = 5$.

The results obtained are not so descriptive, but we considered include them in this work. This results suggest that information is the same for all the cases of degeneracy, redundancy and simple RBNs, thus the complexity reduction of the networks is similar for all cases. In Appendix there is a detailed information of the results.

Chapter 6

Conclusions

We defined three types of degeneracy: *function degeneracy*, *input degeneracy* and *output degeneracy* in RBNs. We performed experiments with these types of degeneracy, and compared them with redundancy as defined in [GKS06] and with simple RBNs.

We performed statistical measures for RBN ensembles with $N = 20$ and $N = 100$ with $K = \{1, 2, 3, 5\}$, which include attractor lengths (A), attractor average lengths (L) and percentage of states in attractors ($\%SIA$). Due to computational constrains, we could not perform these experiments for larger networks. In the same way, we performed experiments to measure the sensitivity to initial conditions, in RBN ensembles with $N = 20$, $N = 100$ and $N = 800$ with $K = \{1, 2, 3, 5\}$. For $N = 20$, 1000 experiments were performed and 100 experiments for other cases.

In natural systems, both redundancy and degeneracy are observed [Wag05a]. Our results suggest that these two properties increase the robustness of RBNs. Further work is required to study the similarities and differences between degeneracy and redundancy.

The effect of degeneracy is stronger on RBNs with a high connectivity K, i.e. in the chaotic regime. RBNs in the ordered and critical regimes ($K \leq 2$) seem not to be affected neither by the addition of any type of degenerate nor redundant nodes. However, in RBN ensembles with $K \geq 2$ the number of attractors is lower for the three types of

degeneracy; also the length of the attractors is shorter compared with a simple RBN. We also found that degeneracy increases the robustness of RBNs accordingly to the experiments performed of sensitivity to initial conditions. This increases the probability of neutrality in their evolution, as changes in the RBN (genotype) have a lower probability of changing its function (phenotype).

These results suggest that degeneracy, as well as redundancy, can facilitate not only robustness, but also evolvability, allowing new functionalities to arise from nodes with small variations either in function or structure without changing too much the dynamics.

The results we present in this work suggest that degeneracy and redundancy promote criticality, which is present in natural networks [AC03, OC06], in what otherwise would be chaotic networks. Preliminary results showed that the effects of redundancy and degeneracy are additive. It would be interesting to study in greater detail different combinations of degeneracy, and whether their effects on criticality are cumulative: for abstract RBNs and for real GRNs.

A drawback of the studies of discrete dynamic networks criticality based on sensitivity to initial conditions is that they restrict the critical regime to a phase transition. Information-theoretical measures [LPZ08, WLP11] might offer an alternative to better characterize the critical regime and the effect of degeneracy and other properties which promote criticality [Gerss].

Appendix A

Numerical Results

In this section we present numerical results of the experiments presented in Chapter 5, which consists of statistical properties of the RBN ensembles and Sensitivity to Initial Conditions. For statistical results we present numerical of the following:

- Attractors A.
- Attractors average length $\bar{L}$.
- Percentage of states in attractors $\%SIA$

for RBN ensembles with $N = 20$ which $N_{deg} = 5$, $N = 80$ which $N_{deg} = 20$. All the experiments for $K = \{1, 2, 3, 5\}$.

For the Sensitivity to Initial Conditions, we present numerical results for RBN ensembles with $N = 20$ which $N_{deg} = 5$, $N = 80$ which $N_{deg} = 20$. All the experiments for $K = \{1, 2, 3, 5\}$.

N_{deg}	ϕ	ι	ω	$\bar{A}$	$\bar{L}_e$	$\%SIA$	σ_A	σ_{L_e}	$\sigma_{\%SIA}$
5	0.0	0.0	0.0	1.667	2.893	2.76E-4	1.446	5.579	5.32E-4
5	0.0	1.0	1.0	1.547	2.605	2.48E-4	1.249	4.896	4.67E-4
5	1.0	0.0	1.0	1.613	2.77	2.64E-4	1.922	8.066	7.69E-4
5	1.0	1.0	0.0	1.515	2.446	2.33E-4	1.148	4.380	4.18E-4
5	1.0	1.0	1.0	1.717	3.131	2.99E-4	2.563	10.577	10.09E-4

Table A.1: Statistical results for case: $N_{core} = 15$, $K = 1$ (for 1000 experiments).

.

N_{deg}	ϕ	ι	ω	$\bar{A}$	$\bar{L}_e$	$\%SIA$	σ_A	σ_{L_e}	$\sigma_{\%SIA}$
5	0.0	0.0	0.0	3.357	11.741	11.2E-4	2.963	15.321	1.46E-3
5	0.0	1.0	1.0	3.092	10.183	9.71E-4	3.079	14.961	1.43E-3
5	1.0	0.0	1.0	3.064	9.095	8.67E-4	2.843	10.970	1.05E-3
5	1.0	1.0	0.0	3.823	8.733	8.33E-4	2.347	12.313	1.17E-3
5	1.0	1.0	1.0	3.317	11.085	10.6E-4	3.216	17.037	1.62E-3

Table A.2: Statistical results for case: $N_{core} = 15$, $K = 2$ (for 1000 experiments).

.

N_{deg}	ϕ	ι	ω	$\bar{A}$	$\bar{L}_e$	$\%SIA$	σ_A	σ_{L_e}	$\sigma_{\%SIA}$
5	0.0	0.0	0.0	4.698	27.830	2.65E-3	3.402	25.171	2.40E-3
5	0.0	1.0	1.0	4.078	17.981	1.71E-3	3.055	14.663	1.40E-3
5	1.0	0.0	1.0	3.942	18.382	1.75E-3	2.721	14.398	1.37E-3
5	1.0	1.0	0.0	3.808	18.328	1.75E-3	2.413	14.396	1.37E-3
5	1.0	1.0	1.0	4.173	20.045	1.91E-3	2.876	15.123	1.44E-3

Table A.3: Statistical results for case: $N_{core} = 15$, $K = 3$ (for 1000 experiments).

.

N_{deg}	ϕ	ι	ω	$\bar{A}$	$\bar{L}_e$	$\%SIA$	σ_A	σ_{L_e}	$\sigma_{\%SIA}$
5	0.0	0.0	0.0	5	188.489	17.9E-3	2.120	135.635	12.93E-3
5	0.0	1.0	1.0	4	72.592	6.92E-3	2.145	50.513	4.82E-3
5	1.0	0.0	1.0	4	67.793	6.47E-3	2.058	43.555	4.15E-3
5	1.0	1.0	0.0	4	87.519	8.35E-3	2.022	55.231	5.27E-3
5	1.0	1.0	1.0	4	85.129	8.12E-3	2.018	50.447	4.81E-3
5	1.0	1.0	1.0	4	85.129	6.87E-3	2.496	47.294	4.51E-3

Table A.4: Statistical results for case: $N_{core} = 15$, $K = 5$ (for 1000 experiments).

.

N_{deg}	ϕ	ι	ω	$\bar{A}$	$\bar{L}_e$	$\%SIA$	σ_A	σ_{L_e}	$\sigma_{\%SIA}$
5	0.0	0.0	0.0	5.881	823.324	7.85E-2	2.209	464.872	4.43E-2
5	0.0	1.0	1.0	5.467	304.344	2.90E-2	2.252	183.516	1.75E-2
5	1.0	0.0	1.0	5.025	208.595	1.99E-2	1.948	117.753	1.12E-2
5	1.0	1.0	0.0	5.111	282.7	2.70E-2	1.980	153.498	1.46E-2
5	1.0	1.0	1.0	5.129	255.066	2.15E-2	1.917	116.750	1.11E-2

Table A.5: Statistical results for case: $N_{core} = 15$, $K = 10$ (for 1000 experiments).

.

N_{deg}	ϕ	ι	ω	$\bar{A}$	$\bar{L}_e$	$\%SIA$	σ_A	σ_{L_e}	$\sigma_{\%SIA}$
20	0.0	0.0	0.0	1.44	2.4	1.118E-7	0.791	2.569	1.196E-7
20	0.0	1.0	1.0	1.39	2.48	1.155E-7	0.968	3.806	1.772E-7
20	1.0	0.0	1.0	1.54	2.74	1.276E-7	1.187	4.168	1.941E-7
20	1.0	1.0	0.0	1.30	2.16	1.006E-7	0.640	2.580	1.201E-7
20	1.0	1.0	1.0	1.43	2.51	1.169E-7	0.875	3.381	1.574E-7

Table A.6: Statistical results for case: $N_{core} = 80$, $K = 1$ (for 100 experiments).

.

N_{deg}	ϕ	ι	ω	$\bar{A}$	$\bar{L}_e$	$\%SIA$	σ_A	σ_{L_e}	$\sigma_{\%SIA}$
20	0.0	0.0	0.0	17.34	203.72	9.49E-6	38.168	584.488	2.734E-5
20	0.0	1.0	1.0	10.53	1253.76	58.4E-6	21.092	11359.171	53.156E-5
20	1.0	0.0	1.0	8.19	88.42	4.12E-6	11.557	248.901	11.641E-5
20	1.0	1.0	0.0	10.81	100.21	4.67E-6	22.733	194.201	9.076E-5
20	1.0	1.0	1.0	13.51	178.63	8.32E-6	22.482	373.029	1.744E-5

Table A.7: Statistical results for case: $N_{core} = 80$, $K = 2$ (for 100 experiments).

.

N_{deg}	ϕ	ι	ω	$\bar{A}$	$\bar{L}_e$	$\%SIA$	σ_A	σ_{L_e}	$\sigma_{\%SIA}$
20	0.0	0.0	0.0	62.25	171397.226	5.90E-3	100.902	303593	14.13E-3
20	0.0	1.0	1.0	16.50	29635.8	1.38E-3	34.726	106242	4.95E-3
20	1.0	0.0	1.0	19.95	43764.52	2.04E-3	48.501	147545	6.87E-3
20	1.0	1.0	0.0	20.71	44957.42	2.09E-3	57.938	175238	8.16E-3
20	1.0	1.0	1.0	39.35	101125.804	4.60E-3	74.093	226109	10.52E-3

Table A.8: Statistical results for case: $N_{core} = 80$, $K = 3$ (for 100 experiments).

.

			avg ΔH				
N	N_{deg}	K	Simple	FD	ID	OD	R
20	5	1	-6.340E-2	-6.361E-2	-6.384E-2	-6.509E-2	-6.376E-2
20	5	2	-4.595E-2	-4.668E-2	-3.115E-2	-5.155E-2	-4.574E-2
20	5	3	3.001E-2	-1.777E-3	-1.050E-3	1.335E-2	4.432E-4
20	5	5	2.481E-1	1.468E-1	1.097E-1	1.669E-1	1.089E-1

Table A.9: Avg ΔH for RBN ensembles with $N_{core} = 15$, $K = \{1, 2, 3, 5\}$ (for 1000 experiments). Simple: Simple RBN, FD: RBN with function degeneracy, ID: RBN with input degeneracy, OD: RBN with output degeneracy and R: RBN with redundant nodes

.

			σ				
N	N_{deg}	K	Simple	FD	ID	OD	R
20	5	1	1.818E-2	1.873E-2	1.855E-2	1.687E-2	2.505E-2
20	5	2	7.574E-2	7.224E-2	7.273E-2	7.074E-2	8.356E-2
20	5	3	1.771E-1	1.535E-1	1.516E-1	1.727E-1	1.638E-1
20	5	5	2.475E-1	2.510E-1	2.403E-1	2.594E-1	2.505E-1

Table A.10: Standard deviations for RBN ensembles with $N_{core} = 15$, $K = \{1, 2, 3, 5\}$ (for 1000 experiments). Simple: Simple RBN, FD: RBN with function degeneracy, ID: RBN with input degeneracy, OD: RBN with output degeneracy and R: RBN with redundant nodes

.

			avg ΔH				
N	N_{deg}	K	Simple	FD	ID	OD	R
20	5	1	-1.250E-2	-1.244E-2	-1.227E-2	-1.250E-2	-1.250E-2
20	5	2	-3.680E-3	-8.353E-3	-9.050E-3	-6.870E-3	-6.011E-3
20	5	3	3.894E-2	5.914E-3	2.118E-3	6.258E-3	4.533E-3
20	5	5	2.817E-1	1.813E-1	1.607E-1	2.432E-1	1.471E-1

Table A.11: Avg ΔH for RBN ensembles with $N_{core} = 80$, $K = \{1, 2, 3, 5\}$ (for 1000 experiments). Simple: Simple RBN, FD: RBN with function degeneracy, ID: RBN with input degeneracy, OD: RBN with output degeneracy and R: RBN with redundant nodes

.

			σ				
N	N_{deg}	K	Simple	FD	ID	OD	R
20	5	1	1.666E-13	8.871E-4	1.863E-3	1.666E-013	1.666E-13
20	5	2	4.227E-2	2.708E-2	2.491E-2	3.287E-002	3.429E-2
20	5	3	1.804E-1	1.571E-1	1.544E-1	1.630E-001	1.653E-1
20	5	5	2.357E-1	2.365E-1	2.296E-1	2.411E-001	2.301E-1

Table A.12: Standard deviations for RBN ensembles with $N_{core} = 80$, $K = \{1, 2, 3, 5\}$ (for 1000 experiments). Simple: Simple RBN, FD: RBN with function degeneracy, ID: RBN with input degeneracy, OD: RBN with output degeneracy and R: RBN with redundant nodes

.

				avg ΔH			
N	N_{red}	N_{deg}	K	Core	R+FD	R+ID	R+OD
20	0	0	1	-6.027E-002	-6.297E-002	-6.443E-002	-6.505E-002
20	5	5	2	-1.127E-002	-5.394E-002	-5.812E-002	-5.350E-002
20	5	5	3	1.188E-001	-2.750E-002	-3.445E-002	-1.190E-002
20	5	5	5	3.470E-001	6.691E-002	1.829E-002	1.121E-001

Table A.13: Avg ΔH for RBN ensembles with $N_{core} = 15$, $K = \{1, 2, 3, 5\}$ (for 1000 experiments). R+FD: RBN with redundancy + function degeneracy, R+ID: RBN with redundancy + input degeneracy and R+OD: RBN with redundancy + output degeneracy

.

				σ			
N	N_{red}	N_{deg}	K	Core	R+FD	R+ID	R+OD
20	0	0	1	3.075E-002	2.281E-002	1.743E-002	2.458E-002
20	5	5	2	1.233E-001	6.024E-002	4.928E-002	7.098E-002
20	5	5	3	2.139E-001	1.229E-001	1.142E-001	1.532E-001
20	5	5	5	2.023E-001	2.262E-001	1.906E-001	2.608E-001

Table A.14: Standard deviations for RBN ensembles with $N_{core} = 15$, $K = \{1, 2, 3, 5\}$ (for 1000 experiments). R+FD: RBN with redundancy + function degeneracy, R+ID: RBN with redundancy + input degeneracy and R+OD: RBN with redundancy + output degeneracy

.

				avg ΔH			
N	N_{red}	N_{deg}	K	Core	R+FD	R+ID	R+OD
20	0	0	1	-1.389E-003	-1.389E-003	-1.389E-003	-1.39E-003
20	5	5	2	5.339E-004	-1.389E-003	-1.138E-004	-1.22E-004
20	5	5	3	2.321E-001	9.903E-002	1.048E-001	9.81E-002
20	5	5	5	4.290E-001	1.048E-001	1.837E-001	1.87E-001

Table A.15: Avg ΔH for RBN ensembles with $N_{core} = 720$, $K = \{1, 2, 3, 5\}$ (for 100 experiments). R+FD: RBN with redundancy + function degeneracy, R+ID: RBN with redundancy + input degeneracy and R+OD: RBN with redundancy + output degeneracy

.

				σ			
N	N_{red}	N_{deg}	K	Core	R+FD	R+ID	R+OD
20	0	0	1	1.339E-014	1.339E-014	1.339E-014	1.34E-014
20	5	5	2	1.347E-002	1.339E-014	1.021E-002	1.09E-002
20	5	5	3	1.887E-001	1.695E-001	1.720E-001	1.69E-001
20	5	5	5	1.489E-001	1.720E-001	2.344E-001	2.35E-001

Table A.16: Standard deviations for RBN ensembles with $N_{core} = 720$, $K = \{1, 2, 3, 5\}$ (for 100 experiments). R+FD: RBN with redundancy + function degeneracy, R+ID: RBN with redundancy + input degeneracy and R+OD: RBN with redundancy + output degeneracy

.

K	Simple RBN	Function Deg	Input Deg	Output Deg	Redundancy
1	0.9895611679	0.9899273337	0.9881121671	0.9885474509	0.9892979959
2	0.8904027185	0.8953620419	0.8936137607	0.8854827164	0.8842975171
3	0.6478774243	0.6586707062	0.6516444379	0.6517814963	0.640998786
4	0.5486910837	0.5505973502	0.5505632245	0.5491972189	0.5448491397
5	0.5210462451	0.5215738776	0.5219414944	0.5212703696	0.51873237
6	0.5098516183	0.5102240872	0.5103267267	0.5099717307	0.5086677228
7	0.50482505	0.5049422296	0.5050946341	0.5048353616	0.5041267551
8	0.5023830548	0.5024592682	0.5025331766	0.5023801169	0.5020009387
9	0.5011914279	0.5012361523	0.5012682962	0.5012203534	0.5009885617
10	0.5005897769	0.5006099417	0.5006469366	0.5006033301	0.5004640253

Table A.17: Mean values for Homeostasis experiments for 1000 RBNs ensembles with $N = 50$, $N_{deg} = 50$ and $K = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$

.

K	Simple RBN	Function Deg	Input Deg	Output Deg	Redundancy
1	0.9864020901	0.9866635907	0.984464443	0.9850168108	0.9855530134
2	0.8137635716	0.8256727399	0.8197283462	0.8096024351	0.8055297582
3	0.2721822287	0.2990287861	0.2831693535	0.2811301765	0.2581777339
4	0.0752040057	0.0783097714	0.0784144597	0.07604513	0.069052685
5	0.0314487134	0.0322055304	0.0327178497	0.0316796118	0.0279259148
6	0.0144710361	0.0149714902	0.0151517638	0.0147016467	0.0127274104
7	0.0070670289	0.0072401978	0.0074386131	0.0071360719	0.0060720579
8	0.0034952339	0.0035784521	0.0036967386	0.0035134334	0.0029568146
9	0.0017736555	0.0018110599	0.0018779887	0.0017820361	0.0014699045
10	0.000905	0.000926	0.000957	0.000914	0.000736

Table A.18: Mean values for self-organization expermients for 1000 RBNs ensembles with $N = 50$, $N_{deg} = 50$ and $K = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$

.

Bibliography

[ABKR07] Maximino Aldana, Enrique Balleza, Stuart Kauffman, and Osbaldo Resendiz. Robustness and evolvability in genetic regulatory networks. *Journal of Theoretical Biology*, 245(3):433–448, April 2007. 30

[AC03] Maximino Aldana and Philippe Cluzel. A natural class of robust networks. *Proceedings of the National Academy of Sciences of the United States of America*, 100(15):8710–8714, July 2003. 62

[AGCK03] M. Aldana-González, S. Coppersmith, and L. P. Kadanoff. Boolean dynamics with random couplings. In Ehud Kaplan, Jerrold E. Marsden, and Katepalli R. Sreenivasan, editors, *Perspectives and Problems in Nonlinear Science. A Celebratory Volume in Honor of Lawrence Sirovich*. Springer Applied Mathematical Sciences Series, 2003. iii, 13, 27

[Ald03] Maximino Aldana. Boolean dynamics of networks with scale-free topology. *Physica D*, 185(1):45–66, 2003. iii, 25, 30

[BABC+08] Enrique Balleza, Elena R. Alvarez-Buylla, Alvaro Chaos, Stuart Kauffman, Ilya Shmulevich, and Maximino Aldana. Critical Dynamics in Genetic Regulatory Networks: Examples from Four Kingdoms. *PLoS ONE*, 3(6):e2456+, June 2008. 30

[Bar02] Albert-Lázló Barabási. *Linked: The New Science of Networks*. Perseus, 2002. iii, 7

[By04] Yaneer Bar-yam. Multiscale variety in complex systems. *Complexity*, 9:37–45, 2004. 58

[Can32] Walter B. Cannon. *The Wisdom of the Body*. W.W. Norton & Company, New York, first edition, 1932. 59

[CMW07] S. Ciliberti, O. C. Martin, and A. Wagner. Innovation and robustness in complex regulatory gene networks. *Proceedings of the National Academy of Sciences*, 104(34):13591–13596, 2007. 2

[Cru94] James P. Crutchfield. Critical Computation, Phase Transitions, and Hierarchical Learning. In Yamaguti, editor, *Towards the Harnessing of Chaos*, Amsterdam, The Netherlands, The Netherlands, 1994. Elsevier Science Publishers B. V. 22

[Dor00] Alan Dorin. Boolean networks for the generation of rhythmic structure. In Brown and Wilding, editors, *Proceedings Australian Computer Music Conference 2000*, pages 38–45, 2000. 14

[DP86] B. Derrida and Y. Pomeau. Random networks of automata: A simple annealed approximation. *Europhys. Lett.*, 1(2):45–49, 1986. 23

[EG01] Gerald M Edelman and Joseph A Gally. Degeneracy and complexity in biological systems. *Proceedings of the National Academy of Sciences of the United States of America*, 98(24):13763–13768, 2001. 3, 4, 32, 33

[FS04] Pau Fernández and Ricard Solé. The role of computation in complex regulatory networks. In Eugene V. Koonin, Yuri I. Wolf, and Georgy P. Karev, editors, *Power Laws, Scale-Free Networks and Genome Biology*. Landes Bioscience, 2004. 20, 30, 33

[Ger02] Carlos Gershenson. Classification of random Boolean networks. In R. K. Standish, M. A. Bedau, and H. A. Abbass, editors, *Artificial Life VIII: Proceedings of the Eight International Conference on Artificial Life*, pages 1–8. MIT Press, 2002. 13, 14, 25

[Ger04a] Carlos Gershenson. Introduction to random Boolean networks. In M. Bedau, P. Husbands, T. Hutton, S. Kumar, and H. Suzuki, editors, *Workshop and Tutorial Proceedings, Ninth International Conference on the Simulation and Synthesis of Living Systems (ALife IX)*, pages 160–173, Boston, MA, 2004. 6, 13, 14, 20, 22, 23, 41

[Ger04b] Carlos Gershenson. Updating schemes in random Boolean networks: Do they really matter? In J. Pollack, M. Bedau, P. Husbands, T. Ikegami, and R. A. Watson, editors, *Artificial Life IX Proceedings of the Ninth International Conference on the Simulation and Synthesis of Living Systems*, pages 238–243. MIT Press, 2004. 46, 49

[Ger05] Carlos Gershenson. RBNLab, 2005. http://rbn.sourceforge.net. 41

[Gerss] Carlos Gershenson. Guiding the self-organization of random Boolean networks. *Theory in Biosciences*, In Press. 23, 32, 62

[GF12] Carlos Gershenson and Nelson Fernández. Complexity and information: Measuring emergence, self-organization, and homeostasis at multiple scales. *Complexity*, page n/a, 2012. 58, 59

[GHA03] Carlos Gershenson, Francis Heylighen, and Centrum Leo Apostel. When can we call a system self-organizing. In *In Advances in Artificial Life, 7th European Conference, ECAL 2003 LNAI 2801*, pages 606–614. Springer-Verlag, 2003. 58

[GK73] L. Glass and S.A. Kauffman. The logical analysis of continuous, nonlinear biochemical control networks. *Journal of Theoretical Biology*, 39:103–129, 1973. 6

[GKS06] Carlos Gershenson, Stuart A. Kauffman, and Ilya Shmulevich. The role of redundancy in the robustness of random Boolean networks. In L. M. Rocha, L. S. Yaeger, M. A. Bedau, D. Floreano, R. L. Goldstone, and A. Vespignani, editors, *Artificial Life X, Proceedings of the Tenth International Conference on the Simulation and Synthesis of Living Systems.*, pages 35–42. MIT Press, 2006. 30, 34, 61

[Gou02] Stephen J. Gould. *The Structure of Evolutionary Theory*. Belknap Press of Harvard University Press, 1 edition, March 2002. 20

[Gro08] Claudius Gros. *Complex and adaptive dynamical systems: a primer*. Springer Complexity. Springer, Berlin, Heidelberg, 2008. 6, 13

[GSFF00a] B. Guo, C. A. Styles, Q. Feng, and G. R. Fink. A Saccharomyces gene family involved in invasive growth, cell-cell adhesion, and mating. *Proceedings of the National Academy of Sciences of the United States of America*, 97(22):12158–12163, October 2000. 32

[GSFF00b] Bing Guo, Cora A. Styles, Qinghua Feng, and Gerald R. Fink. A saccharomyces gene family involved in invasive growth, cellâĂŞcell adhesion, and mating. *Proceedings of the National Academy of Sciences*, 97(22):12158–12163, 2000. 3

[HB97] I. Harvey and T. Bossomaier. Time out of joint: Attractors in asynchronous random Boolean networks. In P. Husbands and I. Harvey, editors, *Proceed-*

ings of the Fourth European Conference on Artificial Life (ECAL97), pages 67–75. MIT Press, 1997. 14

[HMAG02] C. Huepe-Minoletti and M. Aldana-González. Dynamical phase transition in a neural network model with noise: An exact solution. *Journal of Statistical Physics*, 108(3/4):527–540, 2002. 14

[Hop94] J. J. Hopfield. Physics, computation, and why biology looks so different. *Journal of Theoretical Biology*, 171:53–60, 1994. 20

[HSWK02] Stephen E. Harris, Bruce K. Sawhill, Andrew Wuensche, and Stuart Kauffman. A model of transcriptional regulatory networks based on biases in the observed regulation rules. *Complexity*, 7(4):23–40, March/April 2002. 24

[Kau69] S. A. Kauffman. Metabolic stability and epigenesis in randomly constructed genetic nets. *Journal of Theoretical Biology*, 22:437–467, 1969. 13, 14, 30, 41

[Kau93] S. A. Kauffman. *The Origins of Order*. Oxford University Press, 1993. 13, 22

[Kau00] Stuart A. Kauffman. *Investigations*. Oxford University Press, 2000. 22

[KEG02] K. Kappler, R. Edwards, and L. Glass. Dynamics in high dimensional model gene networks. *Signal Processing*, 83:789–798, 2002. 6

[Kim83] Motoo Kimura. *The Neutral Theory of Molecular Evolution*. Cambridge University Press, Cambridge, 1983. 2

[KPS+] Stuart Kauffman, Carsten Peterson, BjÃűrn Samuelsson, Carl Troein, and Lu Tp. Genetic networks with canalyzing boolean rules are always stable. *PNAS*, pages 17102–17107. 24

[Lan90] Christpher Langton. Computation at the edge of chaos: Phase transitions and emergent computation. *Physica D*, 42:12–37, 1990. 22

[LB04] Bartolo Luque and F.J.Fernando J. Ballesteros. Random walk networks. *Physica A: Statistical Mechanics and its Applications*, In Press, 2004. 2

[LPZ08] Joseph T. Lizier, Mikhail Prokopenko, and Albert Y. Zomaya. The information dynamics of phase transitions in random Boolean networks. In Seth Bullock, Jason Noble, Richard Watson, and Mark A. Bedau, editors, *Proceedings of the Eleventh International Conference on the Simulation and Synthesis of Living Systems (ALife XI), Winchester, UK*, pages 374–381, Cambridge, MA, 2008. MIT Press. 62

[LRMC10] Ricardo López-Ruiz, Héctor Mancini, and Xavier Calbet. A statistical measure of complexity. *CoRR*, abs/1009.1498, 2010. 58

[LS97] Bartolo Luque and Ricard V. Solé. Phase transitions in random networks: Simple analytic determination of critical points. *Physical Review E*, 55(1):257–260, 1997. 24

[LS00] Bartolo Luque and Ricard V. Solé. Lyapunov exponents in random Boolean networks. *Physica A*, 284:33–45, 2000. iii, 26, 48

[Mit09] Melanie Mitchell. *Complexity - A Guided Tour*. Oxford University Press, 2009. 1

[NC97] S. Nandi and Parimal Pal Chaudhuri. Reply to comments on "theory and application of cellular automata in cryptography". *IEEE Trans. Computers*, 46(5):639, 1997. 13

[New94] Stuart A. Newman. Generic physical mechanisms of tissue morphogenesis: A common basis for development and evolution. *Journal of Evolutionary Biology*, 7:467 – 488, 1994 1994. 32

[OC06] Panos Oikonomou and Philippe Cluzel. Effects of topology on network evolution. *Nat Phys*, 2(8):532–536, August 2006. 62

[OS02] C. Oosawa and M. A. Savageau. Effects of alternative connectivity on behavior of randomly constructed Boolean networks. *Physica D*, 170:143–161, 2002. 30

[PDFV05] Gergely Palla, Imre Derenyi, Illes Farkas, and Tamas Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435(7043):814–818, June 2005. iii, 8

[PG11] Rodrigo Poblanno-Balp and Carlos Gershenson. Modular random Boolean networks. *Artificial Life*, 17(4):331–351, 2011. 30, 46, 48

[Pob11] Rodrigo Poblanno-Balp. Coupled random Boolean networks and their criticality. Master's thesis, Universidad Nacional Autónoma de México, Ciudad Universitaria, México, 2011. 30

[QNDR03] Tom Quick, Chrystopher L. Nehaniv, Kerstin Dautenhahn, and Graham Roberts. Evolving embodied genetic regulatory network-driven control systems. In W. Banzhaf, Thomas Christaller, Peter Dittrich, Jan T. Kim, and Jens Ziegler, editors, *Advances in Artificial Life: ECAL 2003*, pages 266–277. Springer, 2003. 14

[SCH+04] U. Sauer, F. Canonaco, S. Heri, A. Perrenoud, and E. Fischer. The soluble and membrane-bound transhydrogenases udha and pntab have divergent functions in nadph metabolism of escherichia coli. *J Biol Chem*, 279(8):6613–9, 2004. 33

[Sey94] R. Seydel. *Practical bifurcation and stability analysis: from equilibrium to chaos*. Springer, 1994. 48

[SFAW97] Roland Somogyi, Stefanie Fuhrman, Manor Askenazi, and Andy Wuensche. The gene expression matrix: Towards the extraction of genetic network architectures. *Nonlinear Analysis: Theory, Methods and Applications*, 30(3):1815–1824, 1997. 14

[Sim96] Herbert A. Simon. *The sciences of the artificial (3rd ed.)*. MIT Press, Cambridge, MA, USA, 1996. 2

[SK04] Ilya Shmulevich and Stuart A. Kauffman. Activities and Sensitivities in Boolean Network Models. *Physical Review Letters*, 93(4):048701+, July 2004. 24

[SS96] R. Somogyi and C. A. Sniegoski. Modeling the complexity of genetic networks: Understanding multigenic and pleiotropic regulation. *Complexity*, 1(6):45–63, 1996. 14

[Sta94] Dietrich Stauffer. Evolution by damage spreading in Kauffman model. *Journal of Statistical Physics*, 74:1293–1299, 1994. 48

[Tho78] R. Thomas. Logical analysis of systems comprising feedback loops. *Journal of Theoretical Biology*, 73(4):631–656, aug 1978. 43

[TSE99] Giulio Tononi, Olaf Sporns, and Gerald M. Edelman. Measures of degeneracy and redundancy in biological networks. *Proceedings of the National Academy of Sciences*, 96(6):3257–3262, 1999. 32, 33

[von56] John von Neumann. Probabilistic logics and the synthesis of reliable organisms from unreliable components. In C. Shannon and J. McCarthy, editors, *Automata Studies*, Princeton, 1956. Princeton University Press. 2

[WA96] Gunter P. Wagner and Lee Altenberg. Complex Adaptations and the Evolution of Evolvability. *Evolution*, 1996. 2

[Wag00] Andreas Wagner. The role of population size, pleiotropy, and fitness effects of mutations in the evolution of overlapping gene functions. Working papers, Santa Fe Institute, 2000. 32

[Wag05a] Andreas Wagner. Distributed robustness versus redundancy as causes of mutational robustness. *Bioessays*, 27(2):176–88, Feb 2005. 2, 3, 61

[Wag05b] Andreas Wagner. *Robustness and Evolvability in Living Systems*. Princeton University Press, Princeton, NJ, 2005. 1, 2

[Wag05c] Andreas Wagner. Robustness, neutrality, and evolvability. *FEBS Letters*, 579:1772–1778, 2005. 3

[WB10a] J. M. Whitacre and A. Bender. Degeneracy: a design principle for robustness and evolvability. *Journal of Theoretical Biology*, 263(1):143–153, March 2010. 2, 3

[WB10b] James M Whitacre and Axel Bender. Networked buffering: a basic mechanism for distributed robustness in complex adaptive systems. *Theor Biol Med Model*, 7:20, 2010. 33

[Whi10] James Whitacre. Degeneracy: a link between evolvability, robustness and complexity in biological systems. *Theoretical Biology and Medical Modelling*, 7(1):6, 2010. 1, 2, 3, 32, 33

[WJM10] Bender Axel Whitacre James M. Degeneracy: a design principle for achieving robustness and evolvability. 2010. 1, 30, 33

[WLP11] X. Rosalind Wang, Joseph T. Lizier, and Mikhail Prokopenko. A fisher information study of phase transitions in random boolean networks, 2011. 62

[WP05] Richard A. Watson and Jordan B. Pollack. Modular interdependency in complex dynamical systems. *Artif. Life*, 11(4):445–458, December 2005. 2

[Wue97] A. Wuensche. *Attractor Basins of Discrete Networks*. PhD thesis, University of Sussex, 1997. 20

[Wue98] Andrew Wuensche. Discrete dynamical networks and their attractor basins. In Russell Standish, Bruce Henry, Simon Watt, Robert Marks, Robert Stocker, David Green, Steve Keen, and Terry Bossomaier, editors, *Complex Systems '98*, pages 3–21, University of New South Wales, Sydney, Australia, 1998. 13, 22

[Wue02] Andrew Wuensche. Basins of attraction in network dynamics: A conceptual framework for biomolecular networks, 2002. 41