UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

POSGRADO EN CIENCIA E INGENIERÍA DE LA COMPUTACIÓN

"Coupled Random Boolean Networks And Their Criticality"

T E S I S

que para obtener el grado de:

Maestro en Ciencias
(Computación)

P R E S E N T A

Rodrigo Poblanno Balp

Directores de Tesis:

Dr. Christopher Stephens

Dr. Carlos Gershenson García

México, D.F. 2011

Acknowledgements

I want to thank my family. My sister Naara, my parents Pawa and Miguel Ángel, my uncle Enrique and my cousins Edward and Enriquito without whose support and love this work would have been much tougher.

Special thanks to my supervisor Carlos Gershenson, for his support, courage and advice.

My friends Brian, Mauro, Gaby "Travis", Gabriel "Píter", Óscar, Goussie, Rock, Rodrigo "Villa", Paola, Solie, Abo for all the good moments and memories we have built together.

Also my best friend Laura, with whom I began to build my master's degree and gave me excellent feedback, academically and morally speaking.

My university friends and colleagues, Víctor, Cheko, Etna, Jorge, Tzolkin and Edgar "el Pong".

To my jury, Chris Stephens, Max Aldana, David Rosenblueth and Elena Álvarez, for their advice and help.

To all the professors of the Computer Science programme and to the department's assistants Diana, Lulú and Amalia.

To CONACyT that supported financially this work. CVU-268628.

And last but not least, to the Computer Science master's degree programme for the finacial support to go to Odense, Denmark to present this work in August, 2010.

Contents

Preface

Throughout Chapter 1 we give an introduction to Complex & Dynamical Systems, treating Chaos and Complexity as relevant topics for this work. The basic concepts of Graph Theory, showing its origins from the Eulerian theorems up to a brief explanation of Gene Regulatory Networks which are modelled in this work. The foundations of other types of networks like Small-World Networks and Scale-Free Networks are also exposed in this chapter.

The next chapter deals with random Boolean networks (RBNs), showing the underlying properties of such networks. We start from the primitive and original properties of such networks, and go onto other more specialised networks such as GARBNs or DARBNs. We study the RBN model as preparation for modular RBNs. We then show alternate models, pointing out the main differences in properties among such models.

The third chapter introduces an important topic for this work: Modularity. We explain why modularity has a direct bearing with the work we present and explain the different ways in which modularity is defined. This chapter combined with the previous one will create the basis for modular RBNs (mRBNs). We also show the relevance and the ubiquity of modularity in fields such as molecular biology or sociology.

Coupled Random Boolean Networks (that is, modular RBNs) is the main topic of the fourth chapter. We explain this new model and show the new parameters and their effect on the network's main properties, like criticality. Five case-studies were made with which statistical data was retrieved and analysed by simulating mRBNs. The analyses and comparison of these cases are explained both tabularly and graphically.

Finally the results are discussed and analysed. An explanation as of why those results were obtained is also a subject of this chapter. A brief discussion on the future work and improvements to our model and in the last part we conclude the work.

This work was presented at the twelfth International Conference on the Synthesis and Simulation of Living Systems (ALife XII) which was held in Odense, Denmark in August 2010 [Poblanno-Balp and Gershenson, 2010]. It has also been accepted for publication in the Artificial Life Journal [Poblanno-Balp and Gershenson, 2011].

This work was financially supported by CONACyT CVU-268628 in the masters degree program during the years 2009 and 2010. Special thanks.

Chapter 1

Introduction

This chapter will focus on the foundations and the background of this research work and will establish the main subject. Once the panorama of Random Boolean Networks (RBNs) is open, the importance of their study, and more specifically, the criticality of coupled RBNs shall be clear.

1.1 Dynamical and Complex Systems

Complex systems play a important role in modern scientific approach to solve problems, since such systems have proved better results when the problem landscape is too large (complex) for a typical optimisation technique. When this happens, analytical methods tend to find local optima, which are often distant from the global solution, making an accurate prediction almost impossible [Gershenson, 2009].

1.1.1 Dynamical Systems

There are the so called *non-stationary systems* because the equations that describe the system change along time, thus using an analytical technique may present false positive results or a global optimum valid only in the past. Dynamical and non-stationary systems have been studied by physicists and mathematicians for a long time and solving them usually requires specialised and advanced mathematical techniques. Dynamical systems are based on rules, and the evolution of one of these rules gives notion of the state of the system in a short period of time ahead in the future. Knowing the state of the system for all future time steps requires the system to be *iterated* or *solved*. Once the system is solved it is possible to determine all its possible future states: its trajectory. Numerical methods implemented on computers have helped on determining the trajectories of dynamical systems.

A typical example of this is the motion of the planets. Ancient scientists discovered that keeping a record of the planets' positions helped them predict their future positions. In a more formal sense, the state of a physical system at certain point in time can be represented as a single point in the state space $\mathcal{M}$ and the evolution of any of those points is in fact called *dynamics*; the function f which specifies where the point is at time t is called *evolution rule*. We will call a system *deterministic* if there exists a unique evolution rule f that maps exactly one point in $\mathcal{M}$ to one point in the future. However, this is not always possible to find such a function. For example, knowing the temperature of today is not sufficient condition to know or predict the temperature of tomorrow. An approach to this issue could be the broadening of $\mathcal{M}$ in order to capture many of these points that could help us predict some (less inaccurate) future states.

1.1.2 Chaos

A deterministic system is one whose present state can be completely determined by its initial conditions, implying that there is no randomness involved: the system is not *stochastic*. In a stochastic system the initial conditions determine the future states of the system in a partial manner; hence its future states or trajectories are described in terms of probability distributions. This uncertainty could be due to noise or some other external aspects that we cannot control. Therefore the present state is determined by past initial conditions and noise added during the past evolution.

Complex dynamics inside a deterministic system can dupe us and make us mistake the system as a stochastic one, when it is not. We will call a system *chaotic* if it follows deterministic laws of evolution but whose outcome is highly sensitive to small changes in the initial conditions [Cvitanović et al., 2010]. This is deterministic chaos, but there are also different types of chaos, like non-deterministic chaos, and quantic, to mention a pair. Deterministic chaos often limits a system's predictability due to high sensitivity to initial conditions.

1.1.3 Complexity and Complex Systems

As most complexity texts do, we will remark that there is no unified definition of "complexity", because complexity appears in many different contexts finding its roots in natural and social sciences; thus, picking out a definition of any context somehow clashes with the other definitions. We will take, as F. Heylighen, the etymological definition and compile a more general definition embracing most of the contexts mentioned above [Gershenson, 2008].

In Latin, *Complexus* means "entangled" or "embracing". We could think that something complex requires two or more distinguishable parts, in some way connected to each other so that their separation is difficult to perform. Complexity deals with the entanglement of these distinguishable components, making the whole, complex, and adding several properties (as we will describe shortly) to the system, many of which are *emergent*. This is sometimes referred to as emergent behaviour. We will say a system becomes more complex if the number of distinguishable elements increases and the connections between themselves get more entangled. Notice that some connections could be made with the outer world (environment, feedback, noise, etc.).

As we shall later discuss, there are several levels in which complexity could be found, obliging us to broaden our definition or specify the scale or level we are dealing with. But what are the main differences between simple and complex systems? There are basically three main points which discern simple systems [Érdi, 2008]:

- Single cause and single effect.
- Small changes in the cause imply small changes in the effect
- Predictability

To summarise these three points we could say a simple system does not offer big surprises if we alter it, making the simple system (more) *predictable*. At the same time, complex systems tend to have the opposite properties:

- Small changes in the cause imply big (or vast) changes in the effect
- Unpredictability

- Emergent behaviour

Within complex systems, the effect can serve as both cause and effect, if we see this causal loop as feedback. A question we may ask is what parts of the system do together what they would not do on their own. There are new properties that appear with the mixing of simple properties.

There are at least two kinds of complexities directly linked to randomness [Érdi, 2008]; the *Algorithmic Complexity*, also known as first-class complexity and *Structural complexity*, consequently referred to as second-class complexity. This first-class complexity is the Kolmogorov complexity: as randomness increases, complexity increases. It is related to binary strings (consisting of 0's and 1's), in the sense that the complexity of a random string s is defined by the number of bits of the shortest binary program that describes that string s. In this way, the string 011011011011011011 has an evident 011 pattern inside and an (informal) description could be "repeat 011 six times". The description of the string s is $d(s)$, and its complexity is denoted by $K(s)$. If the string s happens to be completely random[1] we will say it is *maximally complex*, for the shortest binary program that describes it is —at least— as long as the string s itself, because the only pattern we can find in s, is the whole string.

Structural complexity is often referred to as "Complexity between order and randomness", or second-class complexity. It can be shown that for extremely ordered or extremely random systems, complexity is not at its maximal; the maximum complexity is reached at a middle point between randomness and order. Figure 1.1 shows the three cases in which structural complexity may be found.

[1] Assuming there is no way to compress such string.

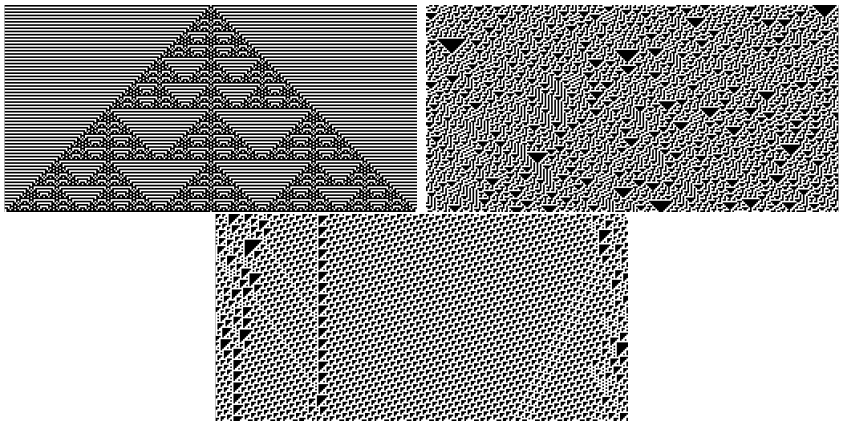

Figure 1.1: Cellular Automata showing purely ordered (top-left) and purely random (top-right) patterns and *class 4 behaviour* (bottom) which is neither completely random nor completely ordered.

1.1.4 Cybernetics

The term Cybernetics comes from the Greek word *kybernetes*, meaning "governor" or "steersman". This term was first used by Plato in ancient Greece and later in the nineteenth century by André-Marie Ampère. Both saw it as the science of effective government. The concept was revived by mathematician Norbert Wiener (1894 – 1964) in his book "Cybernetics: Or the Control and Communication in the Animal and the Machine". Inspired by the current mathematical advances on information theory, Wiener began to develop the theory of organisation and communication in systems [Wiener, 1965]. This includes the study of feedback, black-boxes, an self-organisation. [Wikipedia, 2010]

Among other contributions of Cybernetics, like the explanation of purposiveness, and the goal-directed behaviour, the one that concerns us in this work is *feedback*. Complex systems are perhaps the "New Cybernetics" [Gershenson, 2008] because they also deal with

coordination and control. But how are these systems linked with Complexity?

We know things change in time. With this property in mind, we can expect that to fully understand a system in which many changes occur in time, we may require something additional, as the system itself is more entangled along time. Changes within a system can be random, and as we described earlier in the first-class complexity: as randomness increases, complexity increases too. Nonetheless, the system can grow in complexity if we add the *feedback*, mentioned above. This again refers to the fact that complex systems' output (effect) can serve as both input (cause) and output (effect).

If we imagine a simple system like the one shown in Figure 1.2,

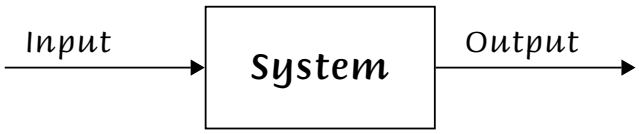

Figure 1.2: Simple system with input and output.

We can anticipate it not to be complex, or at least not as complex as the one in Figure 1.3.

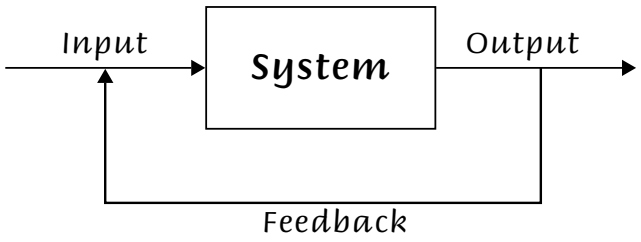

Figure 1.3: System with feedback.

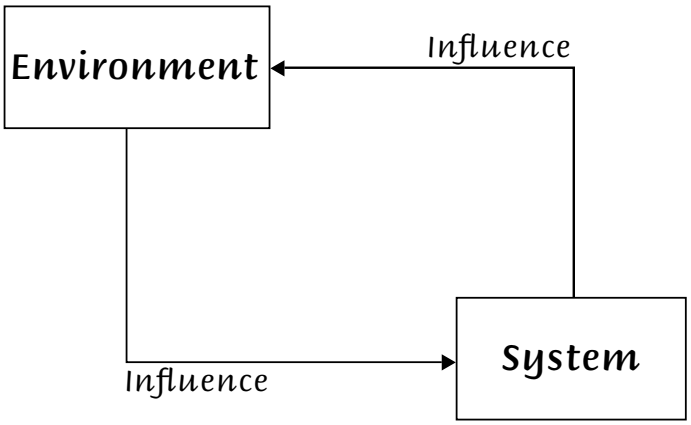

Figure 1.4: Complex and random (from the system's point of view) feedback.

In Figure (1.4), the environment could be seen as a feedback, as the system's output eventually becomes the environment's input, in the same way as the environment's output

is finally the system's input again. Certainly this environment's input could be regarded as *noise* because it would be —from the perspective from within the system— random.

One of the most important studied features about Complex Systems is *robustness*, which deals with the behaviour of the system in order to react favourably to the random external signals, or noise.

1.2 Complexity in Molecular Biology

Genetics is nowadays one of biology's most studied and important fields. Since the father of modern genetics, Gregor Mendel (1822 – 1884), made his inheritance experiments on certain traits of pea plants, the scientific world discovered a whole new scene, but it wasn't until the early twentieth century that scientists redirected their attention towards genetics.

With his two laws, Mendel defined the basic behaviour of a gene, categorising it as "particulate factor" [Lewin, 2004] that passes unchanged from parent to progeny. The central dogma of molecular biology was originated when the gene was discovered as the functional hereditary unit which originates a new product, either RNA or a protein. Among several other problems that needed to be solved and that were gradually solved, the main problem was to determine the relation between both DNA and proteins structures.

Genes are only information, as they do not take any functional part in any hereditary process, instead genes are taken into account as instructions. But this might not be as true as we may think. There has been controversy recently because it has been found that genes do take a fundamental part on cells' molecular functions [Neuman, 2008]. Complexity in

DNA → RNA → protein

Figure 1.5: Schema of the central dogma of molecular biology. Replication copies DNA, which is later transcribed into RNA. The processing phase takes RNA to the cytoplasm for mRNA to carry coded (translation phase) information via ribosomes for protein synthesis.

Molecular Biology is ubiquitous for there are many elements interacting among each other. If we recall Figure 1.4, we saw that this could already be a complex system, now imagine not having only one "feedback" interaction, but thousands or hundreds of thousands. This makes it quite difficult to predict a (biological) system's current state. We have to take into account —not only the elements of the system, as reductionism suggests, but— the high amount of interactions the elements share.

1.3 Networks

Computer Science and Mathematics have —among many others— a special topic in common: the study of graphs. Leonhard Euler (1707 - 1783), a Swiss physicist and mathematician, is reckoned as the father of graph theory; his 1736 paper *The Seven Bridges of Königsberg Problem* is considered the first published paper regarding graph theory [Biggs et al., 1986]. Even though the mathematical problems treated at Euler's time were mainly focused on motion and measurement, he and his contemporaries managed to deviate mathematical studies on the "intuitive" or "puzzle-like" problems of graph theory. Nowadays, graph theory has captured the interest of computer scientists and mathematicians for its transcendental results on non-trivial "everyday problems".

The problem of the Königsberg bridges takes place in the Prussian city of Königsberg (today Kaliningrad, Russia) through which the Pregel river flows as shown in Figure (1.6).

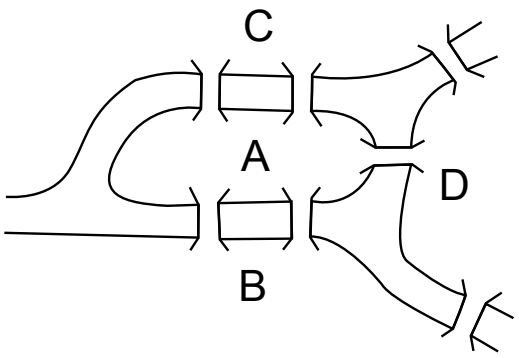

Figure 1.6: Königsberg bridges connecting landmasses. The city consists of the areas marked by A, B, C and D.

The problem was to find a walk through the city crossing each bridge only once[2].

Euler proved that there is no such walk for this problem and that the only relevant characteristic of the walk is the order in which the bridges are traversed. This notion was put in formal (mathematical) terms, by eliminating all features except the landmasses and the bridges connecting them. In a more mathematical point of view, a landmass is converted to an abstract *vertex* (or node) and each bridge is converted to an abstract connection named *edge* as shown in Figure 1.7.

In an intuitive way, a graph is a mathematical model which creates a relation between two

[2]The islands could not be reached by any route other than the bridges, and every bridge must have been crossed completely every time. That is, one could not walk half way onto the bridge and then turn around and later cross the other half from the other side.

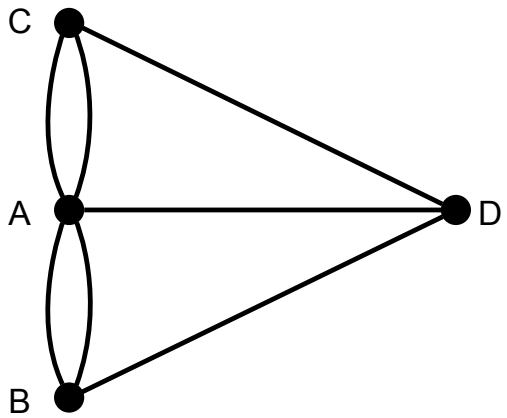

Figure 1.7: Königsberg bridges graph. Landmasses from Figure 1.6 are converted into vertices and bridges into edges.

objects of a collection. In a formal way:

Graph. A Graph is an ordered triple $G = (V(G), E(G), \psi_G)$, where

$V(G)$ is a nonempty set of *vertices*
$E(G)$ a set of *edges* and
ψ_G is the *incidence function* which associates with each edge of G,
an unordered pair of vertices of G.

Then if e is an edge, and u and v are vertices such that $\psi_G(e) = uv$ then e is said to join vertices v and u and both are *ends* of e.

Two ends of an edge are said to be *incident* with the edge, and vice versa. For any pair of vertices u and v which are incident with the same edge, we will say they are *adjacent*. A loop is an edge whose endpoints are the same vertex (node). A graph that lacks loops is called *simple graph* and an edge with distinct ends is called *link*. The *degree* $d_G(v)$ of a

vertex v is the number of edges incident to it[3].

A *walk* in a graph G is a finite non-null sequence $W = v_0, e_0, v_1, e_1 \dots e_k, v_k$ such that for $1 \leq i \leq k$, the ends of e_i are v_{i-1} and v_i; thus W is a walk from v_0 to v_k[4]. As walks are specified by the sequence of vertices, in a simpler form a walk is only specified by the vertices it traverses. Moreover, if the sequence of edges of a walk W are distinct, W is a *trail*; if, besides, the vertices $v_0, v_1, v_2, \dots, v_k$ are distinct, W is said to be a *path*. Finally, two vertices v and u are said to be *connected* if there exists a (v, u)-path.

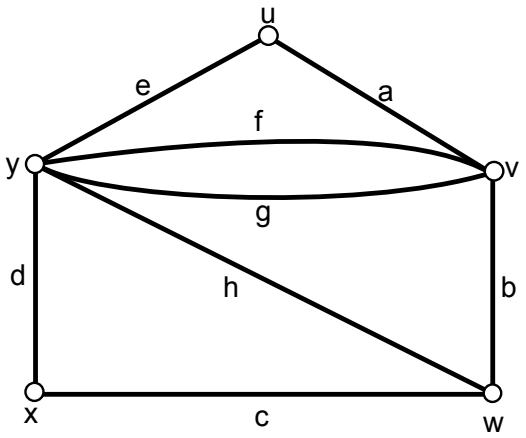

Figure 1.8: Simple graph with 5 vertices and 8 edges. There is a *walk* uavfyfvgyhwbv, that is, a sequence of vertices and edges with no restriction on repetition. A *trail* wcxdyhwbvgy, where no edges are repeated; and a *path* xcwhyeuav where neither edges nor vertices are repeated.

For this work, and for many "everyday problems" such as the routes taken by cars inside a city, graph is not enough. If we had to find the shortest path to go from point A to point B in a city, we must have extra information, for real streets have one-way or double

[3]Each loop counting twice.

[4]Or a $(v_0, v_k) - walk$.

senses, consequently we shall introduce *orientation* to graphs. Many social, technological and natural events are explained using the sort of graphs we are about to explain, so in order to enter the network territory correctly, we shall give a few more definitions.

A **directed graph** or **digraph** is simply: a graph with orientation in edges. Put formally:

Digraph. A Digraph is an ordered triple $D = (V(D), E(D), \psi_D)$,where

$V(D)$ is a nonempty set of *vertices*
$E(D)$ a set of *edges* and
ψ_D is the *incidence function* which associates with each edge of D
an ordered pair of vertices of D.

If e is an edge of a digraph D and u and v are vertices such that $\psi_D(e) = (u, v)$, then e joins u and v; u is then called the **tail** of e and v is called the **head** of e. We shall read this as "the edge e goes from u to v".

Although the current work is about (Random Boolean) Networks, we will not use the strict and formal definition of a Network, which encourages not only oriented but weighed arcs; it also implies having two disjoint sets of vertices. Given that the difference between Networks and Digraphs will not necessary be for the purpose of this work, we will omit the formal definitions and from now on, a network will be a digraph, and vice versa.

Euler showed us that networks have properties, hidden in their construction, that limit or enhance our ability to do things with them [Barabási, 2002]. These properties gave birth to graph theory.

1.4 Small-World and Scale-Free Networks

There are many kinds of networks and all of them have been deeply studied for a over a decade. This section will introduce *Random*, *Small World* and *Scale-Free* Networks to have a full perspective of complex networks and a wider panorama of the Complex Networks field.

1.4.1 Random Networks

Random Networks are generated randomly. They can be characterised by two values, N and z. N refers to the number of nodes and z to the *coordination number*. The *coordination number* is the average number of links per vertex. Alternatively we could define a probability p to find a given link as the *connection probability*. The *connection probability* is the probability that a given edge occurs between two given nodes.

The *network diameter* D is the maximum degree of separation between all pairs of nodes. For a random network with N vertices we have:

$$z^D \approx N, \qquad D \propto \frac{logN}{logz} \tag{1.1}$$

since every node has z neighbours, and z^2 next-nearest neighbours, and so on. The *Path Length* between two nodes is the number of edges on the shortest path between the two nodes. The *average distance* $\mathcal{L}$ of a network is the average of the minimal path length between all pairs of nodes of a network. The *Clustering Coefficient* C is the average fraction of pairs of neighbours of a node that are also nodes of each other; it quantifies how much connected a node is to its neighbours. Suppose a vertex v has k_v neighbours, then at

most $\frac{k_v(k_v-1)}{2}$ [Watts and Strogatz, 1998] arcs can exist between them, this happens when every neighbour of v is connected to every other neighbour of v.

1.4.2 Small-World Networks

As mentioned before in the introduction to digraphs, a good example of a "real" problem was traffic flow in a city; some other good examples would be the Internet (which happens to be a huge network) and an abstract network of the acquaintances in a neighbourhood, a big enterprise, and —why not— the whole world. Surely it has happened to any of us that we meet someone who happens to know (or know somebody who knows) a relatively close friend. When this sort of *coincidences* happen, we often say "What a small world!".

Intuitively one would think that it is easier to be connected with others in a short neighbourhood, or a relatively big company; and the forward thought would be that the number of hops required to get in touch with any other person would rise if we take it to a city or a world-wide level.

In the 1950s, psychologist Stanley Milgram asked himself this question an decided to make an experiment. The experiment consisted of a group of common persons attempting to relay a letter to a distant stranger by giving the letter to an acquaintance and having the acquaintance handing in the letter to one of his or hers acquaintances and so on, until the target stranger received the letter at the end of the chain. In his study, Milgram discovered that in average the number of persons involved through the chain were six.

This phenomenon nowadays is known as "six degrees of separation", for it only takes at most six persons for any two given strangers around the world to reach each other. In their

1998 paper "Collective Dynamics of 'Small World' Networks", Duncan Watts and Steven Strogatz [Watts and Strogatz, 1998] analyse the properties of the small world phenomenon by creating what they called *Small-World Networks*, affirming that the neural network of the *Caenorhabditis elegans* worm, the power grid of western United States and the collaboration graph of film actors were shown to be Small-World networks.

They argued that ordinarily regular or random topologies were assumed, but that many biological, technological or social networks lie between these two extremes, resulting in what today is known as the Watts-Strogatz model. By simply *rewiring* a regular network, it can be transformed into a small world network.

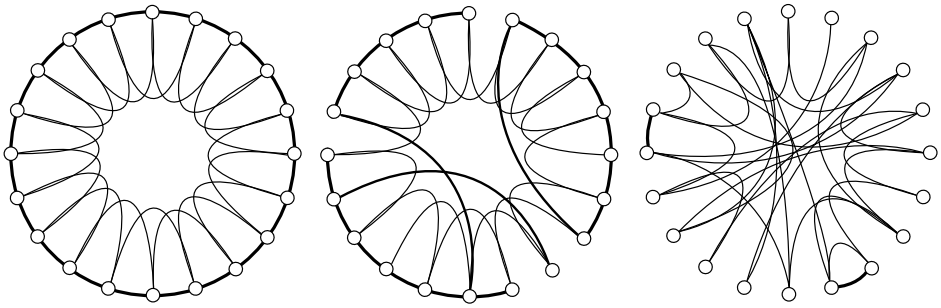

Figure 1.9: A regular graph (left) can be simply rewired with probability p to create a small-world network (centre); as p increases the network becomes more random until it is a random network (right).

Literature often refers to the *small world property*: a network has this property if it has relatively few long-distance connections but has a small average path length $\mathcal{L}$, relative to N. Small-world networks also show a high clustering coefficient.

Consequently, Watts and Strogatz proved that the three very different real world networks (the actors, the power grid and the neural network) had the small world property. The next table shows their results:

	L_{actual}	L_{random}	C_{actual}	C_{random}
Film Actors	3.65	2.99	0.79	0.00027
Power grid	18.7	12.4	0.080	0.005
C. Elegans	2.65	2.25	0.28	0.05

Table 1.1: Actual results of [Watts and Strogatz, 1998]. The table shows both the actual values for *path length* $\mathcal{L}$ and for the clustering coefficient C and the random results.

1.4.3 Scale-Free Networks

Vilfredo Pareto (1848 – 1923) was an Italian economist[5] who wanted to introduce pure mathematics into the economics field, in other words, he wanted economics to be a pure science, just as the beauty of Newtonian physics influenced him. He was the first to determine that income has an 80/20 distribution, or *Pareto Distribution* which is a power law distribution. The 80/20 rules states that 20% of the population controls 80% of the wealth. This rule is not only found in income, but in many other cases like 80 percent of the profits are produced by 20 percent of the employees or 80 percent of Italy's land is owned by 20 percent of the population [Barabási, 2002].

Albert-László Barabási and Réka Albert wanted to determine what was the topology of the WWW. They expected a randomly connected network that should follow Erdős and Rényi theory, which treated random graphs, this because traditionally complex networks had been described that way. What they found was that web pages follow a power law.

A wide variety of complex systems share an important feature: some nodes called *hubs*

[5] As well as engineer, sociologist, and philosopher.

have a big number of connections —hundreds, thousands or even millions of them— while others have just a few, at least compared with hubs. Scale-free networks are robust to random failures, but susceptible to "coordinated" attacks. For these and other reasons scale-free networks have been extensively studied over the past decades.

The basic idea behind Scale-Free networks argues that both theories, Erdős and Rényi's on random graphs and Watts and Strogatz's on Small-World networks, assumed that at all time the nodes in the network were fixed and that the wiring probability rules were random and uniform. Barabási and Albert focused instead on two simple concepts [Barabási and Albert, 1999]:

- Growth
- Preferential attachment

Growth deals with the fact the real-world networks like the WWW are constantly growing (some nodes are even removed) and *preferential attachment* says that a newly connected node would "prefer" to link with a node that has more connections, or in other words, an older node: a hub. Nodes from networks built this way have a power law distribution; which means that the probability $P(k)$ of a node in the network to be connected to exactly k neighbours decays as a power law, following $P(k) \sim k^{-\gamma}$. This is a very interesting result, because these systems obey scaling laws characteristic only of highly interactive self-organised systems and critical phenomena, a feature not found in random network models [Albert et al., 1999, Bunde, 1995].

In order to define intuitively what Scale Free Networks are, we will proceed with an example.

Searching the World-Wide Web a few years ago[6] was a frustrating task, because results were not yielded in an optimal way. If one were to look for "apple records", search engines would give results for pages that had (closely enough) these two words together. It was likely to get the results for a historical list of prices of apples or results for The Beatles recording label. This is why it took so long to find what one really wanted. There were lots of irrelevant results.

We could see the WWW as a network with web pages being nodes and hyperlinks from one page to another being edges. The WWW as many social networks has the structural properties mentioned above: there is a high number of pages with low degree, that is, few in-links; and there is a small number of pages with (very) high degree, that is, many in-links. Also web "communities" tend to have many links among themselves; that is, high clustering coefficient.

To illustrate the main difference between a random graph and a scale-free graph, the following figure is displayed:

The same way, the next figure displays the incoming connections of a given node, versus the frequency of occurrence of such nodes for a scale-free and normal distributions.

What Albert-László Barabási and Eric Bonabeau were trying to do, was to map the Internet, find its diameter and its distribution. They confess that what they really expected was a common Random Network with a typical Normal Distribution with the already studied properties [Barabási and Bonabeau, 2003]. Given the nature of how people build their own web pages, inserting references (hyperlinks) to web pages of their interests and given the diversity of interests and the tremendous amount of web pages they can choose from, they

[6] At least before Google appeared.

expected the resulting network topology to be random. What they actually found was not at all that. There were[7] ≈ 0.001 % of hubs (that is, pages with many incoming references) and $\approx 80\%$ of the pages were "normal" nodes (pages with few incoming references); additionally the distribution found was not Normal, but a Power Law.

*Power Law*s are very different from Normal Distributions. It is also a relationship between two quantities, but here the frequency of an object varies as a power of some attribute of that object. When they counted how many pages had exactly k links, they found that the probability that any node was connected to k other nodes, was proportional to $\frac{1}{k^n}$.

[7]The article [Barabási and Bonabeau, 2003] does not refer to the exact numbers represented by each percentage. Nonetheless estimates of the total number of web pages as of 2008 vary from 100 million to over 10 billion [Mitchell, 2009].

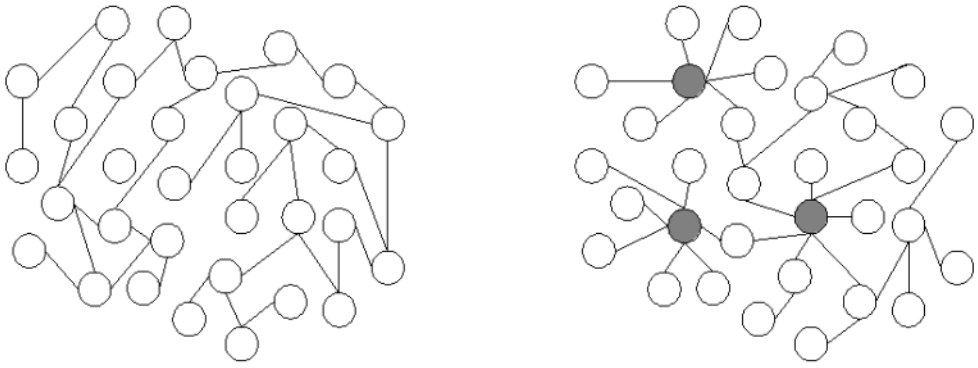

Figure 1.10: A random network on the left and a scale-free network on the right. In the scale-free network, hubs are highlighted. The in-degree for these nodes is higher than for the rest, their number is also lower with respect with the others.

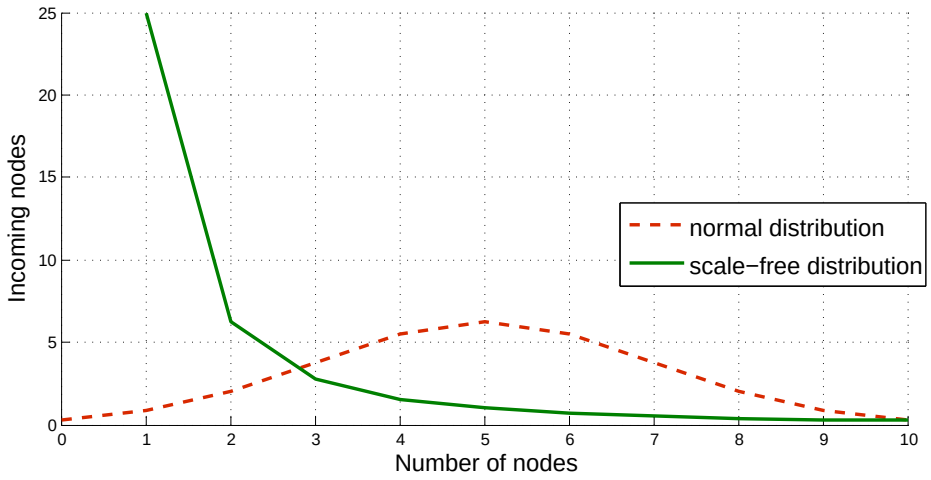

Figure 1.11: Comparison of a normal distribution and a scale-free distribution. For the normal (red and dashed line) most nodes have on average 5 incoming nodes, while for the scale-free distribution, a few amount of them has many connections (hubs) and the rest have few incoming connections.

1.5 Genetic Regulatory Networks

A *Genetic Regulatory Network* (GRN) or *Gene Regulatory Network* is a collection of segments of DNA which indirectly interact with each other through RNA and with other substances in the cell regulating or controlling the rates at which genes are transcribed into mRNA. Each RNA transcript serves as a template for synthesis of a specific protein by the process of translation. An elementary GRN consists of input signalling pathways, regulatory proteins that integrate the input signals, target genes and as a result RNA and proteins produced by the target genes.

A GRN could be viewed as a cellular input-output device. A simple GRN typically consists

of the following elements:

1. The reception of an input signal and a transduction system that mediates intra and extracellular signals.

2. A *core component* complex composed of regulatory proteins and DNA sequences, and

3. Primary molecular outputs from the target genes, which often are RNA or proteins.

The net effects of this process are changes in the phenotype and function. Direct and indirect feedback are important factors of the regulatory process. More realistic GRNs have multiple layers of regulation with outputs of the first layers regulating the expression of another group of genes in a different ulterior layer.

In unicellular organisms regulatory networks react to the environment altering the cell's functions or structures in order to survive and adapt to the environment. The process of wine-making is a clear example of the latter, in which yeast in a sugar solution will switch on genes that convert sugar into alcohol, gaining energy to multiply. Some genes are regulated by a single input mechanism, but, especially in higher organisms, a gene often responds to information from multiple signals via the activity of diverse transcription factors.

Real GRNs are being mapped based on genome-wide expression analyses. Varying widely in number, genomes sizes of living systems can go from as low as 464 genes, for *Guillardia theta* or high as the human genome whose size estimates to be approximately 25,000 genes long. It is worth mentioning that human genome is not one of the biggest genomes sequenced. Nowadays the largest known genome is *Trichomonas vaginalis*. In later sections we will see

the importance of the number of genes in GRNs. By now it is enough just to imagine how complex a network of three thousand nodes could get.

Chapter 2

Random Boolean Networks

Random Boolean Networks —RBNs for short— were proposed by Stuart Kauffman as a model for genetic regulatory networks [Kauffman, 1969] (seen in section 1.5). One of the most important properties of an RBN is that it is *generic*, because few assumptions are made about its connectivity and functionality. Nodes and edges are generated (as its name implies) randomly, which is accurate if the modelled system is highly entangled or highly unknown. In this way, generic properties found in these kind of models can be applied to a particular system to untangle its intrinsic properties.

We have to clarify why this model is Boolean, and that is because in many natural systems interactions are regulated by thresholds. Thus the Boolean model is a good and useful approximation for representing thresholds. Firing potentials in synapses of neurons and activation potentials of chemical reactions in metabolic networks are a pair of examples which could be modelled by the mentioned Boolean approximation. Thus, Boolean states (0 for "off" and 1 for "on") are excellent candidates for modelling thresholds; 0 can be seen

as *below the threshold* and 1 can be seen as *above the threshold*.

The Boolean simplification comes along with another important aspect already treated in this work: complexity. As we saw at the end of the last section, some genomes can be extremely large. To make things simpler, let us take the smallest genome known for an eukaryote at the moment of this writing, *Guillardia theta*, which is only 464 genes long [Douglas et al., 2001]. Using DNA, we have four possible states to combine with: Adenine, Guanine, Thymine and Cytosine; therefore one gene has 4^1 possible values times its number of bases. *Guillardia theta* with its 464 genes has 2^{464} number of combinations for the states of genes, considering two corresponding possible states a gene can be at, either active (1) or inactive (0). This number is immense. Coming back from the numerical parenthesis, that is why having only two possible values *simplifies somehow* the number down to $2^{25,000}$, given that humans have $\approx 25,000$ genes and there are two possible states each gene can be at.

Charles Darwin's original hypothesis did take into account natural selection rules, although the random approach was first taken because little was known about GRNs. It is said to be *random* because we do not know the exact connectivity. Natural selection *is* in fact a random process. Living organisms were therefore generated and improved after many small (random) variations along time.

2.1 The Classical Model

An RBN consists of N nodes (originally called *spins* or state variables, σ_i) which can take a Boolean value, either 0 or 1. Each node has K incoming connections from other nodes,

and with aid of a Boolean function[1], they determine the current state of the node. Both the edges and the logic functions are generated randomly, which means that —in theory— every realisation of the network is different[2]. Every node has a (random) logical function associated with it, and during execution the node's state at a given period of discrete time is defined by the evaluation of the Boolean function, either 0 or 1. These evolve in a synchronous manner according to the transition functions $f_i(\{\sigma\})$. The values taken as input for the Boolean function are the Boolean states of the other K connected neighbours. In the Kauffman model the functions are constrained to depend on —at most— K different input values, chosen randomly:

$$\sigma_i(t) = f_i(\sigma_{\upsilon_i^1}(t-1), \ldots, \sigma_{\upsilon_i^K}(t-1)) \tag{2.1}$$

for every variable (node) σ_i. The input connections to node σ_i are defined in $\{\upsilon_i^1, \ldots, \upsilon_i^K\}$.

Definitions & Properties

In the next section (2.2) we will look at the different types of RBNs and see the updating schemes of each. In the classical model, the updating scheme is *synchronous*, because the states of nodes at time $t+1$ depend entirely on the states of nodes at the previous time step t.

Revisiting the complexity field again, let us calculate how many possible networks are there for a given N and K. First, each node has 2^{2^K} possible logical functions, so for a "typical"

[1] Also called *logical function*.

[2] Almost every realisation is different.

$K = 2$ RBN, each node would have 16 possible Boolean functions. Each node has $\frac{N!}{(N-K)!}$ combinations for K links; again, with $K = 2$ and $N = 10$ the links combination is 90. Furthermore, $\left(\frac{2^{2^K} N!}{(N-K)!}\right)^N$ is the number of all possible networks, given K and N, in this case 1,440. Of course many of these networks would be logically equivalent, nonetheless there are serious computational limitations to run through *all* the possible networks for statistical purposes if we were to use the smallest genome size known to date of living organisms as the value for N.

To begin, a randomly created initial state is chosen and according to this initial state, the dynamics of the net flow and nodes are updated using the Boolean functions. The state space of an RBN is of size 2^N, which means that there is a limited number of states; when the net enters an already visited state, it is said that the net has arrived at an *attractor*. Attractors can be measured by their length, that is, the number of states it has; if it consists of only one state, it is called *point attractor*, or steady state. Whenever it comprises two or more states, it is called *cycle attractor* or state cycle. A set of states that carry the net towards an attractor is called *basin of attraction*.

An easy way to represent the RBN, is by using the Graph Theory traditional method of using a matrix of adjacency. The following is an example of the adjacency matrix of an RBN with $N = 4$ and $K = 2$.

$$\begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \end{bmatrix} \tag{2.2}$$

Reading the matrix by rows, one would say "node 1 (row#1) has an in-link from node 2 (column#2) and node 3 (column#3)". As shown in matrix 2.2, the main diagonal is

zero-valued, that is because there are no loops, although they are allowed.

Order, Chaos And The Edge

Three regimes can be found within RBNs, as well as in many dynamical systems: *ordered*, *chaotic* and *critical*. But how can we identify the regime or phase the net is in? As in cellular automata, we could see the state of nodes along time and determine whether they stay ordered, they change randomly or they stay ordered then go random and come back, that is, somewhere in the edge between order and chaos. This edge is called *critical point* because this is the phase transition, as in fluids dynamics. We could plot in a square lattice all nodes of the net sequentially from initial time t_o to current time t_n, like $t_n, t_{n-1}, \ldots t_1, t_0$. Next time step t_{n+1} would be drawn at the head of the lattice, resulting in $t_{n+1}, t_n, t_{n-1}, \ldots t_1, t_0$. Designating colours would give us an even simpler way to resolve the net's regime. If changing states are coloured with green and fixed states are coloured with red, seas of colours could be watched by letting the dynamics flow. Initially, given that a random state was chosen to begin with, the lattice would show green, but as soon as dynamics stabilised most nodes would be red (static). An ordered regime would appear as a big red *continent* with small green (changing) *seas*. For the chaotic regime the contrary would happen, there would be a big green sea with small red (static) islands. The phase transition between ordered and chaos, "the edge of chaos", occurs when a green sea percolates through the lattice; that is, a big green sea bursts into a big red sea, and vice versa. Figure 2.1 depicts a red sea percolating into a green sea.

Measuring the *stability* of a network is usually something important, as to find out how much damage can be spread and how robust a network is. A way to do this is by randomly

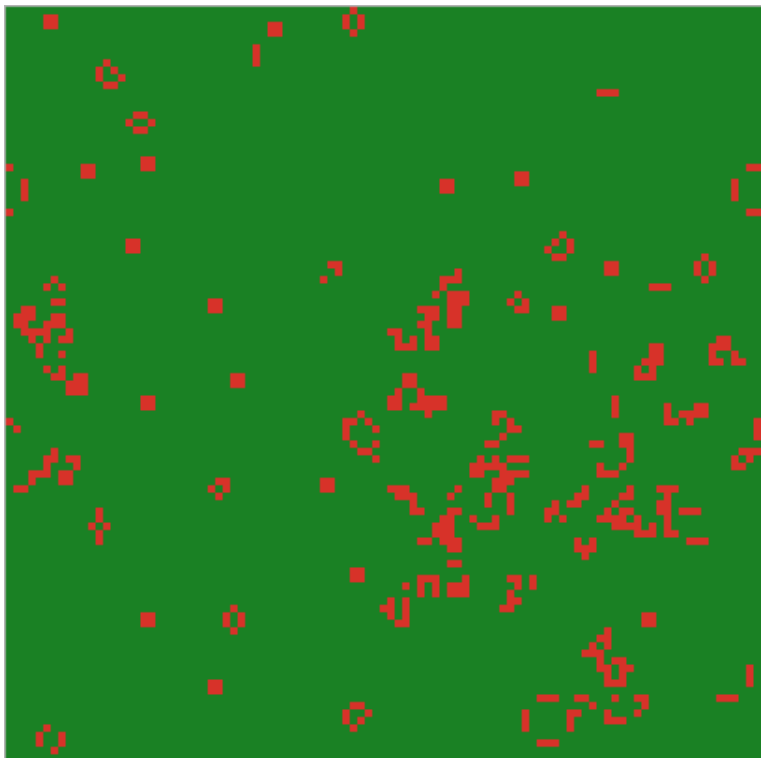

Figure 2.1: Red sea percolating through the green one.

mutating a property of the net, like flipping the state of a node or changing one value from the look-up table. As we know, all nodes are connected by a few hops (as the average path length is low), thus by perturbing a node we can determine how much damage spreads along the net. The lattice mentioned above would show that if the RBN is robust enough, no red sea should turn into green. Nonetheless we could see that just by a little perturbation, a whole red sea could make the net either enter the chaotic regime or be at the edge of chaos. Sometimes a very tiny event can make the dynamics of a system have great consequences or even bring the dynamics to chaos; this is sometimes described as the *butterfly effect*.

Living organisms, computer systems, and in general any system that carries information

from one point in time to a further one, also need stability along time. It has to be strong enough to keep that information. In other words, be robust. But it also has to be flexible enough to allow changes to occur. This thought has conducted researchers to think that life and computation[3] happen at the edge of chaos, as shown with Cellular Automata (CAs) by Langton [Langton, 1990]. In fact there could be systems that can perform the computations even if ordered or chaotic. The former would require much time to do it. The latter would require to be highly redundant to do it [Gershenson, 2004]. We shall remember that redundancy implies robustness.

Phase Transitions, Attractor Lengths And Convergence

Simulations revealed that RBNs moved through the regimes basically when changing K. The actual control parameter is $S = 2p(1-p)K$, but as in this work p remains constant, the control parameter can be seen as solely K. The order parameter is the average hamming distance, which determines, given a control value, which realisations are ordered, chaotic or critical. Ordered regime arises when $K \leq 2$. Chaotic regime takes place when $K \geq 2$.

Hamming Distance. The Hamming Distance δ between two binary strings [4] is the number of bits (or positions) in which both strings differ.

A common way to identify phase transitions in RBNs is to measure the sensitivity to initial conditions, these phase transitions can be numerically-statistically or analytically. Bernard Derrida and Yves Pomeau, were the first to prove analytically the existence of a dynamical

[3] Computation as in biological systems that manipulate information.

[4] The strings do not have to be binary, but Hamming Distance is commonly used in Information Theory, where binary strings are used.

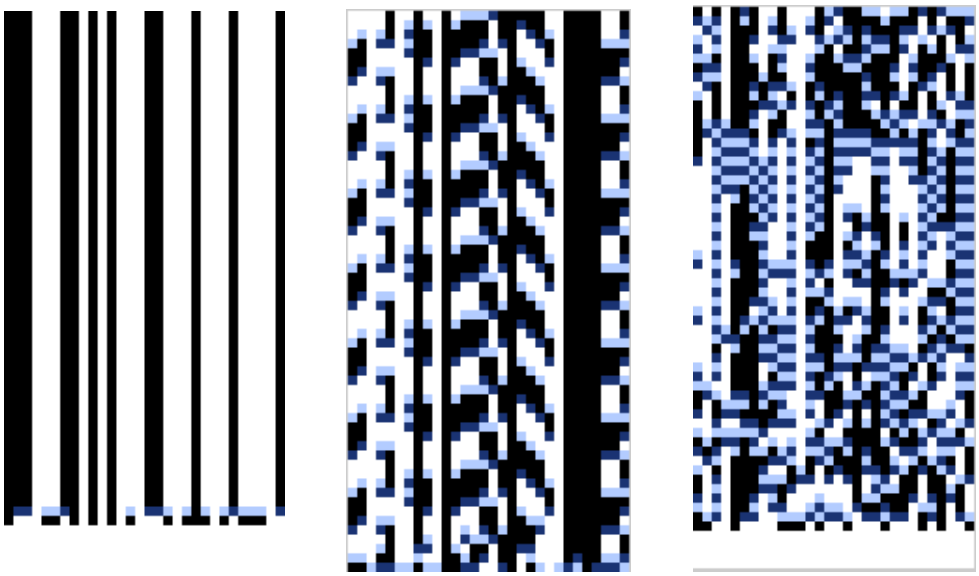

Figure 2.2: Lattices that show the ordered (left), chaotic (right) and critical (middle) regimes. Criticality is reached when $K \geq 2$. The y-axis represents time, with t_0 at the bottom. The x-axis represents the genes.

phase transition controlled by the parameters K and p. For every value of p, there exists a critical value of the connectivity $K_c(p) = [2p(1-p)^{-1}]$, such that if $K < K_c(p)$ all perturbations of an initial state do not propagate (ordered phase). And for $K > K_c(p)$ small perturbations in initial states propagate along the entire system (chaotic phase). Finally the critical region occurs when $K = K_c(p)$. The critical phase of RBNs happens when $K = 2$ [Derrida and Pomeau, 1986] when p is fixed to 0.5. They also presented two variants of the classical RBN model, which will be discussed in section 2.2.

It has been observed the following:

1. For $K = 1$, the probability of having long attractors decreases exponentially and the number of cycles is independent of N. The average length of cycle attractors is of

the order $\sqrt{\frac{N}{2}}$.

2. For $K \geq N$, the average length of attractors grow exponentially. Cycle lengths grow proportional to $2^{\frac{N}{2}}$.

3. In the phase transition, at $K = 2$, both the averages of attractor lengths and the number of attractors grow geometrically with N [Bilke and Sjunnesson, 2001, Socolar and Kauffman, 2003].

High values of N are subjected to numerical analyses but not the whole state-space, for computational limitations; therefore, numerical studies tend to use relatively low values of N, up to approximately 20-30, although with today's computing power and use of computer clusters or multiple-core computers, higher values of N could be tested in order to get results in a couple of weeks or so.

G-Density is one of the parameters used to measure convergence of RBNs, this counts the density of Garden-of-Eden states (GoE). GoE states are those which do not have any ancestors. The dynamics is always irreversible for any state in a basin of attraction. Another measure is the in-degree frequency [Wuensch, 1998] distribution, which can be plotted as a histogram. The in-degree of a state is the number of its immediate predecessors. Convergence in this context refers to the updates required to get to an attractor. Therefore there will be a faster convergence to an attractor in the state-space for states which have a higher in-degree value. In general terms we can state that for high in-degree values it is more likely to be in the ordered regime. There is a higher probability of moving to another basin in chaotic networks, thus low in-degree values tend to diverge. Both G-Density and in-degree frequency distribution reveal the following [Wuensche, 1994, Gershenson, 2004]:

- In the ordered phase there is a very high G-Density and high in-degree frequency; basins are short resulting in high convergence.
- In the chaotic phase there is a relatively low G-Density and a high frequency of low values of in-degrees. Basins are very long thus average transient times are long, which implies low convergence.
- In the critical regime the in-degree distribution resembles a power-law, that is, there are few states with high in-degree value (hubs) and many states with low in-degree value.

2.2 Types Of RBNs

Original RBNs were proposed by Kauffman back in 1969. In this section, we will temporarily rename them to CRBN, for *Classical Random Boolean Network*. We will point out the most important particularities for all kinds of xRBNs.

Derrida & Pomeau [Derrida and Pomeau, 1986] performed an annealed approximation model based on Kauffman's RBN model. They were the first to show analytically that there was a critical connectivity value, K_C. After analysing numerical simulations which showed very different behaviours for $K > 2$ and $K \leq 2$, they sought for the critical K_C to be $2 \leq K_C \leq 3$. Two generalisations were also introduced: one in which K is not (necessarily) the same for all nodes and another which assigned the probability p which governs if values in look-up tables are either one or zero. Variation of network topologies were also studied. Aldana studied free-scale topology for RBNs [Aldana, 2003], based on statistical complex network analyses previously done by Barbási and Albert for the WWW [Albert et al., 1999]

and by Newmann for social networks [Newman, 2001]. Different updating schemes were proposed by Harvey and company [Harvey and Bossomaier, 1997], Di Paolo [Paolo, 2004] and Gershenson [Gershenson, 2002], [Giacobini et al., 2006].

The following subsections briefly describe the most important variations of RBNs and their properties.

DDNs

Discrete Dynamic Networks are the most general and include all the types of networks seen in this section. They were formerly introduced by Andrew Wuensche [Wuensch, 1998] and represent an overall view of networks, because they all have discrete timing, space and values. Networks that, for example, obtain continuous values are studied by Dynamical Systems Theory.

CRBNs

Classical RBNs have been fully described in section 2.1. They are a generalisation for Boolean Cellular Automata, initially developed by John von Neumann in the end of the 1940's decade. Whenever $N = K$, a CRBN is called *Random Map*. Finally we have seen that for different values of (mainly) K and p [Gershenson, 2010, Luque and Solé, 1997], the CRBN can enter any of three regimes: ordered, critical or chaotic, as already mentioned. Given the mentioned features, CRBNs have been widely used to model deterministic systems in many areas of study, from biology and physics to sociology and economy.

Granted that CRBNs are deterministic and that the state space is finite, it can be guaranteed that an attractor will be eventually reached, because once a state reaches an attractor, it will never have states different from the ones in the attractor.

ARBNs

In 1997 Inman Harvey and Terry Bossomaier proposed the *Asynchronous Random Boolean Network*s (ARBNs). They have the same characteristics of CRBNs (2.2) except that the updating of nodes is not only asynchronous but random, that is, at every time step a node is picked randomly to be updated. This makes ARBNs lack cycle attractors. Even so, they have a special type of attractors called *loose attractors* [Harvey and Bossomaier, 1997] which are states that also "grab" the dynamics but the order of these states will not be repeated deterministically, because the updating order is random.

DARBNs

*Deterministic Asynchronous Random Boolean Network*s (DARBN) are basically the same as ARBNs, except that they do not select at random which node to update. Instead, each node has two values associated, p and q ($q < p$). p determines the *period* of an update, i.e. the number of time steps that the node will wait to be updated. q regulates the translation of the update. A node will update whenever $t \bmod p = q$. If a coincidence occurs when two nodes need be updated at the same time, one will be updated after the "other", always considering the resulting network state left by the first. DARBN can be used to model asynchronous phenomena which are not random. They have both cycle and

point attractors.

GARBNs

If we generalise ARBNs, we obtain the *Generalised Asynchronous Random Boolean Network*s (GARBNs). As ARBNs, GARBNs are non-deterministic, which implies they have no cycle attractors, only point and loose attractors. The generalisation is in the number of nodes that can be updated at once. GARBNs pick up a random number of nodes to be updated synchronously. This basically means that GARBNs in a time step can:

- Update no node at all.
- Update one node as ARBNs.
- Update some nodes synchronously.
- Update all nodes synchronously, as CRBNs.

Or in other words, m nodes out of a total of N will be updated at each time step, where $0 \leq m \leq N$.

DGARBNs

Just as we did to convert ARBNs to **D**ARBNs, GARBNs can be converted to **D**GARBNs: *Deterministic Generalised Asynchronous Random Boolean Network*s. We also introduce the parameters p and q, which serve the same as in DARBNs (period and translation).

Opposed to DARBNs, when two (or more) DGARBNs nodes are set to update in the same time step, they will be updated synchronously, i.e. they will be updated at time $t+1$ taking into account the net's state at previous time step t; nodes will be updated sequentially. Cases exist in which DGARBNs and DARBNs overlap. This could happen if only one node is updated at a given time step, e.g. $N = 2$ with one node updating at even time steps $(p = 2, q = 0)$ and the other updating at odd intervals $(p = 2, q = 1)$.

Figure 2.3 shows a map of containment of RBNs [Gershenson, 2002].

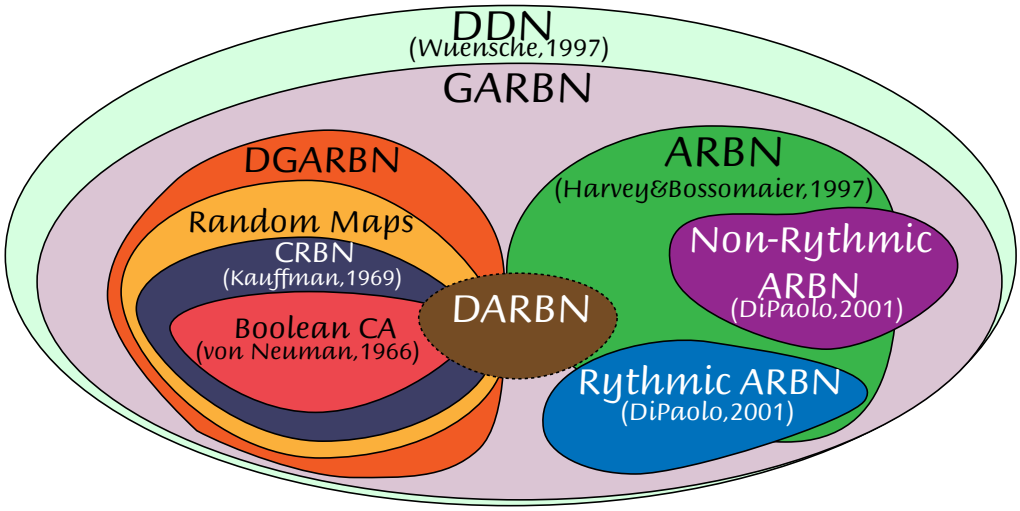

Figure 2.3: Classification of Random Boolean Networks.

DDNs contain all other RBNs, being GARBNs the most general. All others are particular cases of them. Adding to GARBNs parameters p and q, we will have DGARBNs (deterministic). When $N = K$, Random maps emerge (all nodes having $p = 1$ and $q = 0$) and they can be any CRBN with redundancy added. CRBNs are a subset of Random maps. CA are specific cases of CRBNs, where the connectivity is limited by the spatial

organisation of the nodes. Limiting the updating scheme of GARBNs to only one node at each time step, we obtain ARBNs. ARBNs that are deterministic, are in fact DARBNs. These ARBNs have two special cases: with rhythmic and non-rhythmic attractors. DiPaolo [Di Paolo, 2001] defined the measure of rhythmic behaviour as a measure of how patterns occurring at different instants in the history of a system relate to one another.

RBN type	**Synchronicity**	**Determinism**
CRBNs	synchronous	deterministic
ARBNs	asynchronous	non-deterministic
DARBNs	asynchronous	deterministic
GARBNs	semi-synchronous	non-deterministic
DGARBNs	semi-synchronous	deterministic

Table 2.1: Updating schemes of RBNs.

In fact, there are no synchronous & non-deterministic RBNs. The ones that have semi-synchronous update (GARBNs & DGARBNs) are so called because they either behave synchronously or asynchronously, in some cases; in general this means that some nodes are updated synchronously, while the rest are update asynchronously.

Given the characteristics shown for RBNs, there is a peculiarity about point attractors: they are the same for any type of RBN. If we change the updating scheme from an ARBN to a CRBN and we have a point attractor, it will remain the same. This happens when all nodes rules determine that after being updated, their values will remain the same. Consequently, it is unimportant the order in which they update and the number of flipped values: the net's general state will not change.

Chapter 3

Modularity

This chapter is a "bridge" between what random Boolean networks are (2) and what Coupled Random Boolean Networks are(4). It briefly explains why modularity is so important in today's complex systems and the advantages of seeing a system as a group of parts that *interact* among them and with themselves to accomplish a specific —and sometimes very complex— task.

At a very general level, *modularity* is a property a system can have that describes to what extent it can be separated and put back together. But it also refers to the tightness of coupling between its components. Modularity implies "hidden" rules that permit or deny the system's components to be mixed [Schilling, 2000]. Studies of modularity have had an important advance in several branches of biology and cognitive psychology [Fodor, 1983].

It has been observed that most complex systems have a hierarchical structure. That is, a complex system has (complex) subsystems, which have complex subsystems and so forth,

that interact with each other and *execute a discernible function at that level.* The (observed[1]) frequencies associated with these interactions drop as we go up one level in the hierarchy and become less evident, that is why some interactions (hence modularity) are more easily distinguishable, like those of planets, in contrast with those of atomic particles as quarks, which given the speeds are hard to observe. In this sense we can call these *nearly decomposable* (ND) systems [Callebaut and Rasskin-Gutman, 2005, Simon and Ando, 1961].

Just as nature has "the edge of chaos" as a middle point between information loss (chaotic regime) and time loss (ordered regime) in order to optimise evolution, nature has also chosen modularity as means for optimising[2] *something.* But what is that something? What does a system win when it "becomes" modular? It should be clear that is not[3] a matter of understanding, because being able to physically (or abstractly) separate components within a system helps us see the bigger picture. Modularity is an abstract concept that tries to capture the various levels and kinds of heterogeneity [Wagner et al., 2007]. *Heterogeneity* in this sense refers to specialisation, a refinement in a specific task that a module is expected to execute. A module is integrated into a (biological) system with respect to a process —useful for both, the entire system and the module— with certain autonomy of the entire system and other modules. Evidence of modularity highly depends on the scale at which we are observing. Modularity has been studied in protein-protein interactions, gene regulatory networks and variational modularity [Wagner et al., 2007].

Topologically speaking, modularity has some of the already mentioned properties of Scale-Free Networks (see 1.4): hubs. Han, Bertin and collaborators [Han et al., 2004] found dynamically organised modularity in the yeast protein-protein interaction network, and

[1] Interactions do not disappear when we switch levels (modules).

[2] Although it is said that nature does not optimise, but rather finds sufficient solutions.

[3] Or is it?

identified two types of hubs. *Party hubs*, which interact with most of their neighbour[4] hubs simultaneously and *date hubs*, which bind their different hub-neighbours at different times or locations. *In-silico* and *in-vivo* studies support the model of dynamically organised modularity in which date hubs organise the proteome connecting modules to each other, whilst party hubs function inside modules.

Although modularity does not *directly* alter the environment, nonetheless it does contribute directly to the individual's fitness. Nonetheless, evolution has preserved modularity. Good examples would be gene networks involved in development [Solé et al., 2002] and the segment polarity genes of Drosophila, where Dassow, Meir and company suggest that such segment polarity genes are in fact the modules [von Dassow et al., 2000, von Dassow and Odell, 2002]. In their conclusive results in [Solé et al., 2002], Solé et al. conclude[5]:

> Patterns [modules] are common. *In spite of the random character of the network wiring, a large fraction of systems displays spatial structures.* This has non-trivial consequences for the understanding of the evolution of development: our analysis shows that spatial patterns are rather easy to be found and not restricted to small spots in parameter space.

This is an important point, because it states the fact that the topology of the network *really* affects what modularity *actually* is, and consequently its function or purpose; that, without opposing to Han's results [Han et al., 2004].

[4] Originally called *partners*, changed here to neighbour to keep Network notation.

[5] Italics were added to emphasise.

Conservation of modularity is ascribable to *special* features of these modules which are tightly linked to their robustness under different sources of noise [Képès, 2007]. Thus, *neutral models* can be presented in order to justify the fact that natural selection doesn't play but a secondary role in the evolution of modules. This is what Solé calls *Modularity "for free"* [Solé and Fernández, 2003], referring to a mutational process which creates modular structures lacking the involvement of natural selection. The first of these such models is *Duplication-Differentiation* (DD), the second *Neutral Modular Restructuring*. We will briefly describe the former.

The Duplication-Differentiation model consists of networks in which nodes are proteins, and edges protein interactions. A node is randomly picked and duplicated, the chosen protein acquires its 'parent' properties, i.e. its interactions. Provided that proteins have the ability to interact with other proteins by means of their structure without change in gene duplication, this assumption is valid. Next, either with probability δ several of the new protein's interactions are deleted or with probability α a new interaction is added. This model does not specify the rates of δ and α, which means both are free parameters. The model does not explain how protein-protein interaction networks evolve in such manner that lead to modular topological settings. The authors suggest two options for this to happen. First, because sparse networks are favoured by natural selection, for very dense networks can have instabilities leading to chaotic dynamics. Thus by tuning the average degree, selection could reach a stable and robust network with proto-modules [Solé and Fernández, 2003]. The second option is that natural selection directly favours emergent modules to support cellular functions, as if proto-modules had been natural selection's initial target of selecting a sparse graph. Neither possibility fully explains modular structure. Thus, both, the mutational process and natural selection (as a *tuner*) favour modular structure in the DD process. Modularity can be seen as a solution picked and used by Nature over and over

again. This implies that in biological systems solutions are no reinvented, but rearranged in some way that become a new (and possibly a better) solution.

3.1 Weak Links

Péter Csermely gives a very wide perspective of how weak links influence modularity and its relation to Small-Worldness, Scale-Freeness and Nestedness of Complex Networks [Csermely, 2009]. Long-range contacts of small-world networks are formed by weak links. In fact, weak links are required to form modules in proteins which are provided by water molecules; in societies as superficial acquaintances and in cells as protein bridges [Csermely, 2001], thus the creation of modules.

In order to have a notion of what a weak link is, we will expose Berlow's definition [Berlow, 1999]:

Weak Link. A link is defined as *weak* when its addition or removal does not change the mean value of a target measure in a statistically discernible way.

These weak links serve as a buffer to cope with noisy inputs to the network. They also create a resistance against cascading failures, or as we have seen it, damage spreading. Modules have some level of isolation and identity from the rest of the network, nonetheless modules are integrated among themselves; this can only be achieved by (inter-modular) weak links. As the formation of modules implies modularity and modularity implies more stability[6], weak links are essential for stability in a complex network. Having a large number

[6]This is actually shown until the end of next chapter.

of weak links in a complex network avoids wide damage spread and integrates the network as a whole, making the network stable. In contrast to strong links, the removal of a weak link does not necessarily imply a change of parameters, although the network could become less stable [Csermely, 2009].

In the next chapter we will see that weak links are introduced to the classical RBN model and study the implications of this action.

Chapter 4

Coupled Random Boolean Networks

This chapter will deal with the model of *Coupled Random Boolean Network*s (CRBNs) or as we may also call them *Modular Random Boolean Network*s (mRBNs). We will show the details for its construction and the results obtained once simulations were run.

In past chapters we've discussed the critical region of complex networks. This region is also known as the *edge of chaos* and has peculiarities worth studying and analysing. One of the most important singularities it has, is the possibility of preserving information, giving the RBN —or the living system from a biological point of view— the possibility of evolving.

Criticality may be seen as a middle point between adaptability and robustness [Aldana and Cluzel, 2003]. The former must happen to integrate the outer influence into the system in order to adapt to the possible changes occurring in the environment, and the latter to defend itself from bogus signals which could damage the system's integrity. Having found the phase transition, results suggest that evolvability arises precisely at this edge.

We have seen that biological systems tend in a "natural" way to this modularity, thus the immediate question to ask would be «Do mRBNs behave the same way as RBNs?». In other words, how does K influence damage spread, robustness and evolvability in mRBNs.

In this work we put several coupled networks together and show how modularity influences evolvability, by detecting the phase transition between order and chaos: the *edge of chaos*. The model of coupled networks can be viewed as a simplified model of a living system [Villani et al., 2006]. As we know, modularity plays an important role in evolution, for even unicellular organisms have separable functional systems [Wagner et al., 2007] which are relatively autonomous. Coupling of RBNs can be seen as a living system interacting with the environment or as a cell interacting with other cells. The importance of modularity has been in play for quite a long time, for it makes (complex) systems *decomposable* for analysing [Callebaut and Rasskin-Gutman, 2005].

Another of the interesting points of this work is the balance reached in the net after evolution, showing how it could be broken, left intact or slightly modified depending on how *modular* the system is. One should expect that more networks within the ordered regime would imply an ordered *meta*-net, and that more networks within the chaotic regime would force a chaotic *meta*-net.

For now we will not try to quantify modularity. We shall leave the results to speak by themselves and tell us how to quantify it; which, according to the chapter focused on it (3), is not an easy task.

Related Work

In this section we will expose some of the work done in coupled (random) networks, show some of the differences between our model and their basic results. Ho, Hung and Jiang [Ho et al., 2005] studied the dynamics of two coupled random Boolean networks using stochastic coupling techniques. The density evolution of the coupled networks was precisely described by two coupled polynomial maps. Interested mainly on the co-evolution of the two coupled networks and their synchronisation, they showed that the density evolution of the pair of networks was accurately described by the deterministic polynomial maps, providing a way for analytical calculations of networks and also predictability for real networks. They argue that due to simulations taking too long, the coupling had to be restricted to only two coupled networks, although future work plans on doing this for more coupled nets. The latter work was inspired by Andrecut's work [Andrecut, 2005] which also shows that the dynamics of two networks coupled stochastically can be described as two polynomial maps. Results suggested an agreement between the map model and real RBNs.

A slightly different model was proposed by Serra, Villani et al [Serra et al., 2008, Villani et al., 2006] which constructs a 2D CA each cell containing an RBN. This model can be viewed as a simplified model of tissue or a monoclonal colony. Each RBN within a CA's cell is *influenced* by the state of other cell's RBN. They studied the system's response to perturbations and showed how the interactions (and the amount of them) affect the dynamics and the global degree of order.

Finally Hung et al [Hung et al., 2006] coupled two RBNs not using the site-by-site or the all-to-all typical approaches. Instead they link the cells (nodes) of an RBN with those of

another with probability ρ; the node at each end is arbitrarily chosen. They also show that the density evolution[1] of the two RBNs can be described by polynomial maps and they show that complete synchronisation occurs when the coupling parameter $\varepsilon \geq 0.18$, providing good predictions and analytical calculations for actual networks.

Throughout the next section the differences between the mentioned models and our model will be pointed out. In one way all complement each other, in another way each model supports the previous one.

4.1 Definitions & Properties

The mRBN model is a superposition of the (classical) RBN model, which had two main parameters K and N that represented the net's connectivity and the node count, respectively. As we are coupling networks (making a modular meta-network), we will speak of the *meta*-net as the set of all networks seen as one, as a big RBN.

In this model two parameters are added, M and L. M denotes the number of modules the *meta*-net has ($M > 0$)[2]; L is the modular *inter-connectivity* ($L \geq 0$), i.e. the links among modules; these links can also be seen as weak links (as discussed in 3.1). This number is independent of K but is in strong relation to damage spreading and dynamic behaviour, just as K is. Opposed to Csermely's definition, in our context, weak links will not be undirected edges of a graph, they are directed edges instead.

[1] That is, measuring the evolution of density maps for Random Boolean Networks, as described in [Andrecut and Ali, 2001].

[2] It could be seen as the number of RBNs the *meta*-net has.

The updating scheme used in this work is the synchronous one, updating nodes in a sequential form. Let n_i^m be the i-th node of module m, then the updating sequence would be $(\underbrace{n_1^1, n_2^1, \ldots n_n^1}_{\text{Module 1}}, \underbrace{n_1^2, n_2^2, \ldots n_n^2}_{\text{Module 2}}, \ldots\ldots \underbrace{n_1^n, n_2^n, \ldots n_n^n}_{\text{Module n}})$.

Borrowing matrix notation from RBNs[3], Figure 4.1 shows the adjacency matrix for a CRBN with $N = 3$, $K = 2$, $M = 2$ and $L = 1$.

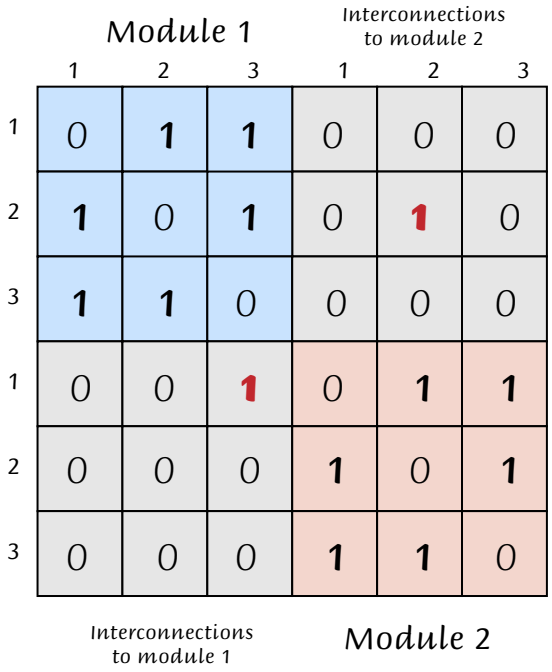

Figure 4.1: Connection matrix for CRBN $N = 3$, $K = 2$, $M = 2$ and $L = 1$. Lack of connections shown in light face. Connections within modules (intra-modular) shown in bold face, and inter-modular connections shown in ultra-bold face and red. The first 3×3 matrix (in light blue) shows connections for module#1, to its right (in light gray), module#1's inter-connections are shown. In this case node#2 receives an inter-modular connection from module#2's second node. The same way, the fourth 3×3 matrix (in light red) shows module#2's connections. To its left, inter-modular connections are shown (in light gray); in this case, module#2 receives a link at node#1 from module#1's third node.

[3]Which in turn borrowed it from Graph Theory.

To simplify, N_{TOT} will denote the total number of nodes, and is given by $N \cdot M$, while the total number of connections is given by:

$$T = M \cdot (K \cdot N + L) \tag{4.1}$$

The average connections per node:

$$K_{\text{TOT}} = \frac{T}{N_{\text{TOT}}} \tag{4.2}$$

Which can also be seen as:

$$K_{\text{TOT}} = K + \frac{L}{N} \tag{4.3}$$

The node-module rate μ is:

$$\mu = \frac{N}{M} \tag{4.4}$$

And finally, two probability measures to be used in the experiments. The following is the probability a given link is **intra-modular**:

$$\kappa = \frac{K}{K_{\text{TOT}}} \tag{4.5}$$

And the probability a given link is **inter-modular**:

$$\lambda = 1 - \kappa \tag{4.6}$$

κ indicates the probability a link is within a module. That is, it should connect a pair of nodes u and v that are part of the same module. Concerning Csemerly's point of view (weak vs. strong links), links in the κ probability are to be considered *strong* links. If we have two nodes u_i and v_j which share an edge e, both nodes corresponding to module i and j respectively, and forcing $i \neq j$, we shall say e is a weak link, because it also connects two nodes, but from different modules.

4.2 Experiments

In order to answer the questions mentioned above, we used the RBNLab, originally developed by Carlos Gershenson. It was modified to use mRBNs and the latest version is hosted at http://rbn.sourceforge.net/.

For the two experiments that will be described in sections 4.2.1 and 4.2.2, five different sets of parameters were used to compile statistically significant data to analyse. These are the five sets:

1. $K = L, \quad \frac{N}{M} \to 1$
2. $K = 1, \quad \frac{N}{M} \to 1$
3. $L = 1, \quad \frac{N}{M} \to 1$
4. $M = 1 \quad N = 20, \quad L = 0 \quad \text{and} \quad K = K_{\text{TOT}}$
5. $N = 1 \quad L = K, \quad M = 20$

Analyses were made comparing sets 1, 2 and 3 which together are also called **κ-exploration cases**; and sets 4, 5 and 1 which together are also called **μ-exploration cases**. In the κ-exploration cases μ is fixed and in the μ-exploration cases, κ is fixed[4]. For all experiments $N_{\text{TOT}} = 20$, where $N_{\text{TOT}} = N \cdot M$. Given that for the experiments N and M were fixed to 5 and 4 respectively, for cases 1 through 3, $\frac{N}{M} = \frac{5}{4} = 1.25$

Case 1 can be seen as a balance between intra-modular links (K dependent) and inter-modular links (L dependent). Case 2 fixes the intra-modular links to 1 ($K = 1$), and

[4]Except for case 1, which also has μ fixed

increments the inter-modular ones. Case 3 instead explores the intra-modular link effect, keeping $L = 1$. As their grouping suggests, the first three cases explore the relation between intra- and inter-modular links, mainly κ.

Next case, case 4, is actually a "typical" RBN. The number of modules is one ($M = 1$). This RBN has twenty nodes all linked with K while $L = 0$, i.e. there are no inter-modular links (since there are no modules). The fifth case goes to the other end, making nets with twenty modules of only one node; most of the intra-modular links will connect the same node (module), causing all the modules to have many auto-links[5]. While analysing the results, we will point out the consequences of having many auto-links.

4.2.1 Sensitivity to initial conditions

One way to obtain random Boolean network's general properties is to measure damage spreading, that is the Sensitivity To Initial Conditions (STIC). In previous sections we have seen that one desirable property for complex networks is their robustness. By measuring damage spreading we can find out how sensitive the network is to small changes. Initial conditions (or initial states) are usually altered to quantify the spread of damage. All initial states are typically generated randomly, the network is run under the conditions established by the initial state and the randomly generated Boolean functions, then the network evolves under those conditions, dynamics flow and eventually the RBN reaches an attractor. As it has been mentioned in previous chapters, the RBN's dynamics highly depend on its topology and on how sparse connections are. The question for the first experiment is how damage scatters when the RBN is modular, and how does modularity

[5] Also called *loops*.

influence this spread.

Let s_i be a random initial state and MAX[6] be the maximum number of time steps the network's dynamics will be executed; then let s_f be the final state the network reaches after MAX steps. We can "save" s_i and generate another "random" initial state s'_i by flipping the k-th bit in the initial state and letting the network run also MAX steps using s'_i as initial state. The RBN will eventually arrive at a final state, s'_f. After both executions are completed, we shall compare both s_f and s'_f and see how much difference there is between them by applying a simple binary function called *Hamming Distance* which was defined in chapter 2.

For example, suppose we have two binary strings $u = 1\mathbf{0}01\mathbf{0}$ and $v = 1\mathbf{1}01\mathbf{1}$, then the hamming distance is $\delta(u, v) = 2$, because u and v have two different values in positions 2 and 5 (in bold).

After having computed $\delta(s_f, s'_f)$ we know how much damage spread along the net, for we knew that $\delta(s_i, s'_i)$ was 1, as we only flipped one bit. Thus, we could estimate how much information or damage is spread by obtaining the difference, $\Delta H = \delta(s_f, s'_f) - \delta(s_i, s'_i)$ and then normalising it, so that $\Delta H \in [0, 1]$. For networks in the ordered regime, $\Delta H < 0$ since similar states converge, that is, the final hamming distance is less than the initial one; whereas for networks in the chaotic regime $\Delta H \to 0.5$, that is, if ΔH went as chaotic as possible, it would approximate $\frac{N_{\text{TOT}}\text{-}1}{N_{\text{TOT}}}$, but this would imply that both trajectories are anti-correlated, which is almost impossible if the networks were generated randomly.

Repeating this and using different parameters for the mRBNs, we are able to judge in which way damage spread depends in ensembles of networks. We shall say that the only

[6] MAX = 10,000 steps

truly critical nets are those whose $\overline{\Delta H} = 0$, and for all the networks with greater values we shall still call them chaotic, but some will be closer to the critical regime than others.

[1101.1893]

N	**K**	**M**	**L**	**T**	$K_{\mathbf{TOT}}$	$\overline{\Delta H}$	μ	κ	σ
5	0.833333	4	0.833333	20	1	-0.0196	1.25	0.83333	0.0128
5	1.666666	4	1.666666	40	2	-0.0033	1.25	0.83333	0.0293
5	2.5	4	2.5	60	3	0.0284	1.25	0.83333	0.0430
5	3.333333	4	3.333333	80	4	0.0879	1.25	0.83333	0.0565

Table 4.1: STIC results for case 1: $K = L$, $\quad \frac{N}{M} \to 1$

Results shown in table 4.1 display the case where $K = L$ (Case 1). There is in each case an increment in K_{TOT}. For each experiment, values for N and K were fixed, as well as T, therefore making K equal to L and solving L from equation 4.1, the appropriate value for L was obtained, and with it, K's value too. σ is the standard deviation.

The two negative values indicate a net in which no damage spread was transferred, while positive values, or values near to zero describe a behaviour tending to the chaotic, because small changes in initial states were more spread. As this is a balanced case (in all ways), μ and κ remained at fixed values. And for the first three cases —the κ-exploration cases— μ will remained fixed.

The case in table 4.2 shows how the net became more critical in a middle point of κ, when $\kappa = 0.33333$. That is, it became more critical when the probability of a link of being intra-modular was (relatively) low. It is also shown how once it became too low, the net became more chaotic.

N	**K**	**M**	**L**	**T**	$K_{\mathbf{TOT}}$	$\overline{\Delta H}$	μ	κ	σ
5	1	4	0	20	1	-0.0191	1.25	1	0.0127
5	1	4	5	40	2	-0.0040	1.25	0.5	0.0324
5	1	4	10	60	3	0.0456	1.25	0.33333	0.0550
5	1	4	15	80	4	0.1156	1.25	0.25	0.0567

Table 4.2: STIC results for case 2: $K = 1$, $\frac{N}{M} \to 1$

N	**K**	**M**	**L**	**T**	$K_{\mathbf{TOT}}$	$\overline{\Delta H}$	μ	κ	σ
5	0.8	4	1	20	1	-0.0189	1.25	0.8	0.0154
5	1.8	4	1	40	2	-0.0053	1.25	0.9	0.0257
5	2.8	4	1	60	3	0.0160	1.25	0.93333	0.0343
5	3.8	4	1	80	4	0.0406	1.25	0.95	0.0444

Table 4.3: STIC results for case 3: $L = 1$, $\frac{N}{M} \to 1$

For the case of Table 4.3 the most critical net occurred when K_{TOT} was 3. This is a case in which most of the links are intra-modular, with some inter-modular (weak) links. Comparing directly this case with the one on table 4.2, we see the effects of modularity, as this case has an even lower $\overline{\Delta H}$. Modularity also made ordered nets get closer to $\overline{\Delta H} = 0$, although the difference between regular RBNs is small and standard deviations relatively high, we can see its effects.

As we can see for all the tables, the first two values of K_{TOT} converge to the ordered regime, suggesting that modularity still requires more links to transfer information.

In general, the STIC results (4.1 - 4.5)show the following:

N	**K**	**M**	**L**	**T**	$K_{\mathbf{TOT}}$	$\overline{\Delta H}$	μ	κ	σ
20	1	1	0	20	1	-0.0222	20	1	0.0094
20	2	1	0	40	2	-0.0054	20	1	0.0320
20	3	1	0	60	3	0.0471	20	1	0.0557
20	4	1	0	80	4	0.1163	20	1	0.0588

Table 4.4: STIC results for case 4: $M = 1 \quad N = 20, \quad L = 0 \quad$ and $\quad K = K_{\mathrm{TOT}}$

N	**K**	**M**	**L**	**T**	$K_{\mathbf{TOT}}$	$\overline{\Delta H}$	μ	κ	σ
1	0.5	20	0.5	20	1	-0.0163	0.05	0.5	0.0151
1	1	20	1	40	2	-0.0029	0.05	0.5	0.0251
1	1.5	20	1.5	60	3	0.0118	0.05	0.5	0.0385
1	2	20	2	80	4	0.0296	0.05	0.5	0.0449

Table 4.5: STIC results for case 5: $N = 1 \quad L = K, \quad M = 20$

- Negative values indicate convergence, or ordered regime.
- Values close to zero indicate the net is in the critical region.
- Values that are above zero indicate chaos.

Although the three last statements are true in general, the cases shown in the past five tables don't show chaotic behaviour, as positive values are really close to zero, thus in a critical region. This suggests that these mRBNs are somehow more robust than traditional ones, as damage is not widely spread. In this sense, robustness means that perturbations do not affect the module's functions. That is why the most robust networks are the ones that almost do not have dynamics (ordered). Chaotic networks, on the other hand are the

least robust, being fragile to perturbations.

As we can see in table 4.4, where $L = 0$ (Case 4), meaning no modularity, when the number of total links is increased to 80, the network's dynamic was chaotic. Nonetheless, results displayed in table 4.5 show that even for higher values of total links T, the nets stayed at the edge, suggesting that modularity makes the network's dynamic stay at the edge of chaos. Results from table 4.5 were *anomalous* because those nets have many self-links, i.e. actual T is smaller. The most representative case is 4.3, i.e. where modules are more isolated.

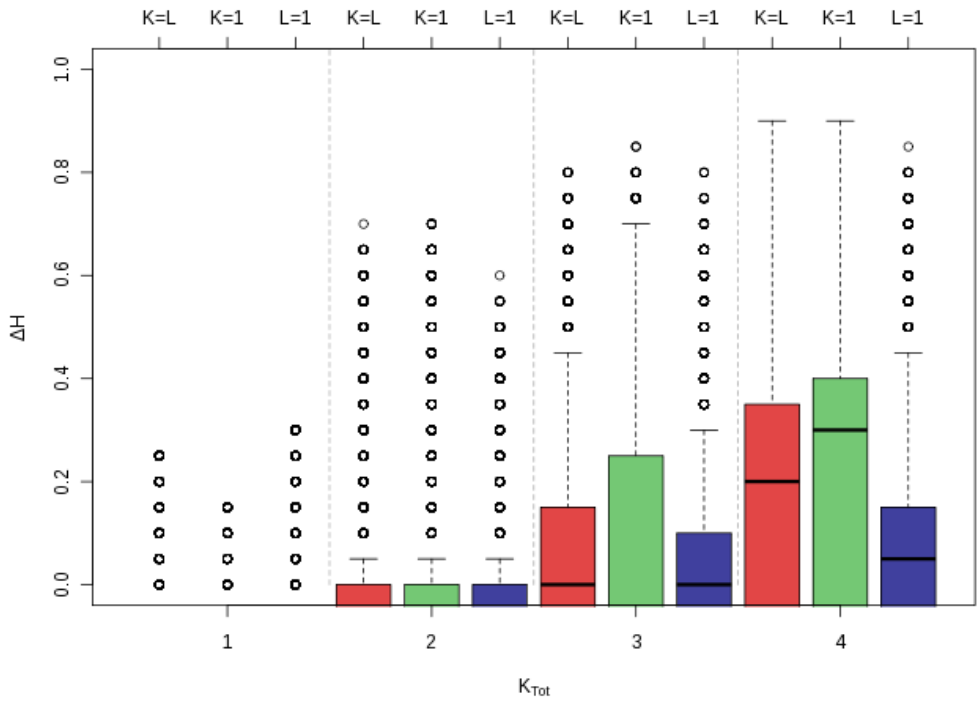

Figure 4.2: STIC results for the κ-exploration cases 1, 2 &3 ($K = L$, $K = 1$ and $L = 1$, respectively).

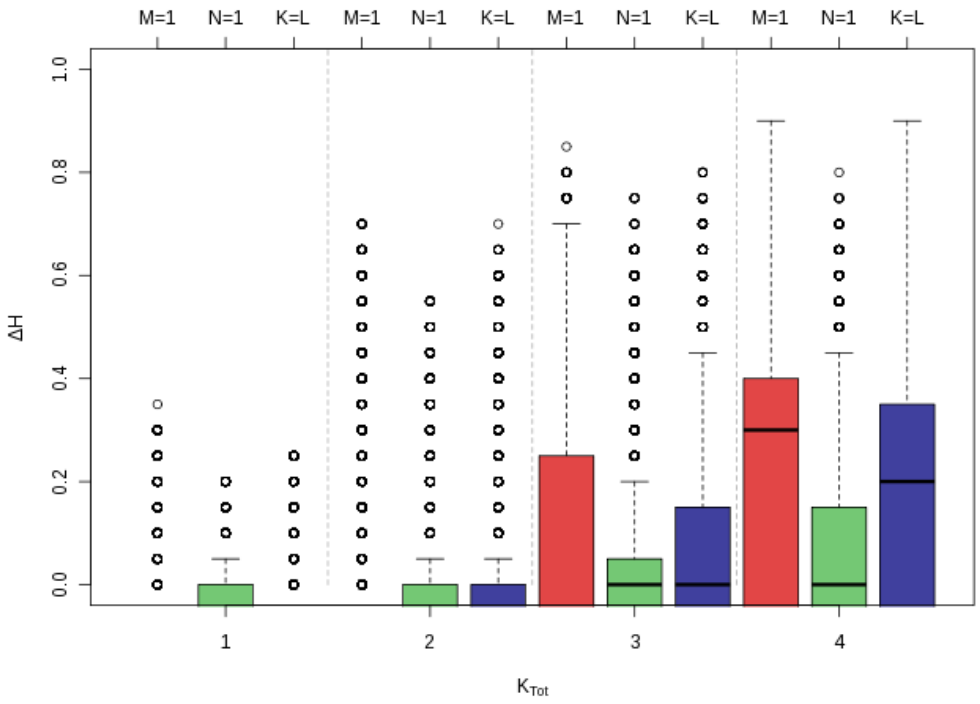

Figure 4.3: STIC results for the μ-exploration cases 4,5 &1 ($M = 1$, $N = 1$ and $K = L$, respectively).

Figure 4.2[7] shows the mRBN dynamics flow towards the ordered regime (maximum stability) in those cases for which $K_{\text{TOT}} = 1$, because the $\overline{\Delta H}$ value goes under zero, denoting convergence. Interestingly, $\overline{\Delta H}$ is zero for all cases in $K_{\text{TOT}} = 2$ and for cases 1 and 3 in $K_{\text{TOT}} = 3$ the mean value stays at zero, making the coupled nets dynamics' flow in the critical regime. Since modules are more heavily connected for $K_{\text{TOT}} = 2$ in case 2 (K=1), it could be expected that damage is more easily spread from module to module, giving the coupled nets a slightly more chaotic behaviour. For $K_{\text{TOT}} = 4$ for the third case (L=1), the overall behaviour is less chaotic, as opposed to cases 1 (K=L) and 2 (K=1), in which chaotic

[7] A full explanation of every component of the plots is available in appendix B.3.

dynamics appear: having less inter-modular links reduces the probability that damage will spread to the other modules. Comparing cases 2 and 3 in $K_{\text{TOT}} = 4$, we can also see the consequences of inter-modular links; case 2 is the most chaotic of all.

In Figure 4.3 we can see a good comparison of the typical RBN (case 4, leftmost) and the modular one (case 5, middle) with almost only auto-links. Although for both in $K_{\text{TOT}} = 4$ there exist some notches close to 1 (too chaotic) the mean value of case 5 is zero[8]. This is because case 5 contains almost only auto-links, thus having one or twenty auto-links does not alter Boolean rules; comparing cases 4 and 5 to case 1 (balance of inter-modular vs. intra-modular links) we see that modularity favours the balanced one, bringing the dynamics to the critical regime.

To have a balance between statistical significance and network size, and to confirm previous results for smaller mRBNs, a new set of experiments of sensitivity to initial conditions was added. Now N_{TOT} was fixed to 400 nodes. The results for cases one, three and four are displayed in the next tables: Table 4.6 confirms the results presented in table 4.1, although

N	**K**	**M**	**L**	**T**	$K_{\mathbf{TOT}}$	$\overline{\Delta H}$	μ	κ	σ
20	0.95238	20	0.9524	400	1.0	-0.0001133	1	0.95	3.32E-05
20	1.90476	20	1.9048	800	2.0	-0.0000160	1	0.95	2.28E-04
20	2.857142	20	2.8571	1200	3.0	0.0017246	1	0.95	2.17E-03
20	3.80952	20	3.8095	1600	4.0	0.0109056	1	0.95	4.69E-03

Table 4.6: STIC results for case 1: $K = L$, $\frac{N}{M} \to 1$ with $N_{\text{TOT}} = 400$ nodes.

these results show smaller numbers and more reliable standard deviations. In this case, the

[8]For all K_{TOT}s except 1!

balance between modules and nodes was achieved.

N	**K**	**M**	**L**	**T**	$K_{\mathbf{TOT}}$	$\overline{\Delta H}$	μ	κ	σ
20	0.95	20	1	400	1.0	-0.0001054	1	0.95	9.51E-05
20	1.95	20	1	800	2.0	-0.0000173	1	0.975	1.61E-04
20	2.9500	20	1	1200	3.0	0.0004838	1	0.98333	5.59E-04
20	4.0	20	1	1600	4.0	0.0016927	1	0.9875	2.34E-03

Table 4.7: STIC results for case 3: $L = 1$, $\frac{N}{M} \to 1$ with $N_{\text{TOT}} = 400$ nodes.

The results from table 4.7 are analogous to those previously shown in table 4.3. The mRBN with $K_{\text{TOT}} = 3$ is closer to the critical region, while the first two cases for $K_{\text{TOT}} = 1$ and 2 the dynamics lay in the ordered regime; being more distant from zero, we can interpret case with $K_{\text{TOT}} = 4$ being chaotic.

N	**K**	**M**	**L**	**T**	$K_{\mathbf{TOT}}$	$\overline{\Delta H}$	μ	κ	σ
400	1	1	0	400	1.0	-0.000122	400	1	1.73E-17
400	2	1	0	800	2.0	0.000081	400	1	5.38E-04
400	3	1	0	1200	3.0	0.007626	400	1	4.53E-03
400	4	1	0	1600	4.0	0.015411	400	1	3.97E-03

Table 4.8: STIC results for case 4: $M = 1$ $N = 20$, $L = 0$ and $K = K_{\text{TOT}}$ with $N_{\text{TOT}} = 400$ nodes.

Finally this last table shows results for a big "normal" RBN, just as table 4.4 did. These results confirm that networks with $K = 2$ as analytically proved by Derrida and Pomeau, lie on the critical regime: the value is very close to zero, compared with the other values of K_{TOT}. In table 4.4 the closest value to zero was indeed the one for $K_{\text{TOT}} = 2$, but it converged as $\overline{\Delta H}$ was negative. Here this value diverges staying very close to zero. The

standard deviation is low too, while with the smaller RBN (N_{TOT}=20) it was higher than the averaged value, which made it a very unreliable result.

The graph comparing cases 3 and 4 is displayed next: Values for $K_{\text{TOT}} = 1$ and $K_{\text{TOT}} =$

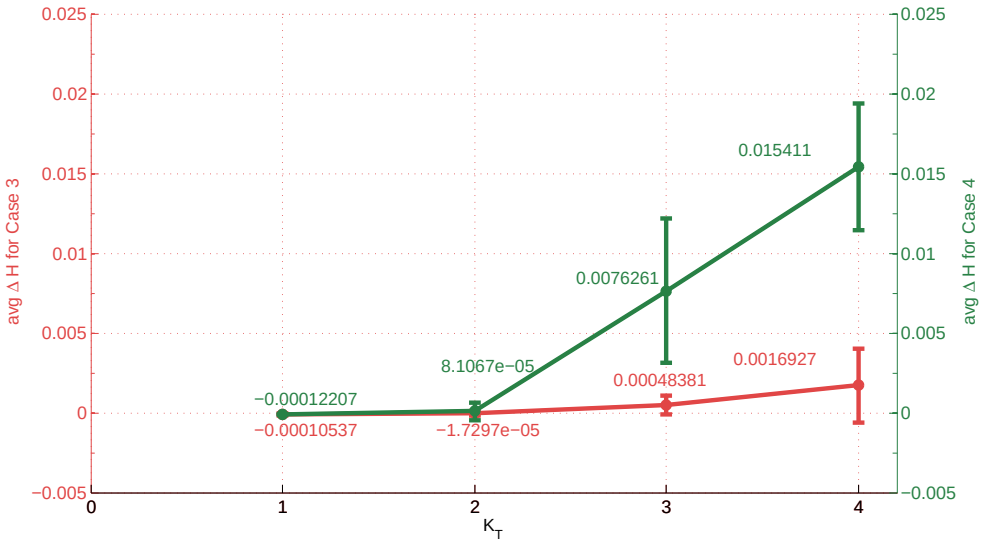

Figure 4.4: Comparison of case 3 (red) and case 4 (green) of STIC results for the new cases where N_{TOT} = 400.

2 remain almost the same for both cases, all negative except for case 4 with $K_{\text{TOT}} = 2$. For the typical RBN (case 4) with $K_{\text{TOT}} = 4$ the average hamming distance grows much compared to the modular RBN (case 3), whose value, while actually chaotic, lies much closer to zero. The big difference is shown for $K_{\text{TOT}} = 3$, because the mRBN value is in the critical region (very close zero), while for the normal RBN the value is one order higher.

4.2.2 Statistics

There are several measurable properties in mRBNs additional to $\overline{\Delta H}$, which only told us how damage was spread inside the net. As mentioned in section 2, all RBNs have at least one attractor, thus the **Number of Attractors** of a net is a good thing to count. As we will show later, there exists a relation between the attractor count and the regime in which the dynamics of the network lie. All attractors consist of a set of states through which the net's dynamics run. The length of this sequence is called **Attractor Length**.

Of all the 2^N states contained in the state-space —which is indeed a number that grows fast— we could see how many of them are contained in attractors. We will call this the **Percentage of States In Attractors** or **%SIA** for short.

These three properties are measured in the Statistics experiment series, and exactly as the STIC experiment (4.2.1) required an upper bound in the number of time steps the net would run through, the Statistics experiment requires a limit too. In this case, this limit is variable. We will not enter in details[9] as of why and when this limit is modified; but it is mainly because as N grows, things get tougher for the computer. Statistics experiments for nets of $N \approx 50$ nodes would require weeks to finish.

The Stats experiments were applied to the same five cases used in the STIC experiments. For the next tables, *Le* stands for Lengths and, A for the Number of Attractors. T values are 20, 40, 60 and 80, respectively, and K_{TOT} runs from 1 through 4.

In table 4.9 we can see that the number of attractors considerably gets higher when $T = 60$. Suddenly when $T = 40$ or $T = 80$ the number of attractors is lower. The result for $T = 80$

[9] See appendix B for a more detailed explanation.

N	**K=L**	**M**	**T**	$K_{\mathbf{TOT}}$	**A**	**Le**	**%SIA**	μ	κ	σA	σL	$\sigma \%SIA$
5	0.83333	4	20	1	4.03	2	0.0009	1.25	0.83333	5.722	1.435	0.0019
5	1.66666	4	40	2	12.65	4	0.0051	1.25	0.83333	16.073	3.619	0.0083
5	2.5	4	60	3	18.06	7	0.0106	1.25	0.83333	18.618	6.140	0.0122
5	3.33333	4	80	4	14.10	13	0.0125	1.25	0.83333	14.789	13.468	0.0129

Table 4.9: Stats results for case 1: $K = L$, $\frac{N}{M} \to 1$

is expected to have a bigger average attractor length, because the number of total links is high. But comparing %SIA for $T = 60$ and $T = 80$ we see that there is not much difference.

N	**K**	**M**	**L**	**T**	$K_{\mathbf{TOT}}$	**A**	**Le**	**%SIA**	μ	κ	σA	σL	$\sigma \%SIA$
5	1	4	0	20	1	5.38	2	0.0013	1.25	1	8.063	1.848	0.0025
5	1	4	5	40	2	4.89	3	0.0016	1.25	0.5	5.045	3.012	0.0026
5	1	4	10	60	3	4.66	7	0.0028	1.25	0.33333	3.583	7.573	0.0029
5	1	4	15	80	4	4.39	22	0.0079	1.25	0.25	2.312	26.448	0.0080

Table 4.10: Stats results for case 2: $K = 1$, $\frac{N}{M} \to 1$

Having a very low intra-module connectivity ($K = 1$) we can see in table 4.10 that the number of attractors almost remains the same for any T. The number of attractors even gets its maximum value at $T = 20$, although the attractor length is quite low.

There is an abrupt jump in the number of attractors in table 4.11 when T goes from 20 to 40. Although table 4.3 did not show that jump in damage spread, we can assume that an attractor is reached very early in the execution, thus because of the considerably low connectivity both in K and L, the mRBN remained fairly ordered. In the remaining results for T higher than 20, even though the *meta*-net connectivity could induce chaos, modularity

N	**K**	**M**	**L**	**T**	$K_{\mathbf{TOT}}$	**A**	**Le**	**%SIA**	μ	κ	σA	σL	$\sigma\%SIA$
5	0.8	4	1	20	1	3.45	2	0.0007	1.25	0.8	4.155	1.508	0.0013
5	1.8	4	1	40	2	15.77	5	0.0071	1.25	0.9	20.528	3.792	0.0123
5	2.8	4	1	60	3	29.30	10	0.0231	1.25	0.933333	28.898	10.294	0.0338
5	3.8	4	1	80	4	34.36	14	0.0376	1.25	0.95	31.973	14.615	0.0432

Table 4.11: Stats results for case 3: $L = 1$, $\frac{N}{M} \to 1$

lowers the damage, keeping the three cases closer to criticality.

N	**K**	**M**	**L**	**T**	$K_{\mathbf{TOT}}$	**A**	**Le**	**%SIA**	μ	κ	σA	σL	$\sigma\%SIA$
20	1	1	0	20	1	1.68	2	0.0003	20	1	1.667	1.997	0.0011
20	2	1	0	40	2	3.15	3	0.0010	20	1	3.371	2.762	0.0016
20	3	1	0	60	3	4.23	8	0.0026	20	1	3.563	9.586	0.0027
20	4	1	0	80	4	4.43	22	0.0081	20	1	2.407	23.662	0.0078

Table 4.12: Statistics results for case 4: $M = 1$ $N = 20$, $L = 0$ and $K = K_{\mathrm{TOT}}$

The cases run in table 4.12 are RBNs, rather than mRBNs, because there is only one module and the inter-modular connectivity (L) is zero. The number of attractors remains almost even, discarding $K = 1$ where the net is ordered, the remaining three cases go chaotic.

These last results show the opposite case to the one shown in table 4.9. Here all modules contain only one node, thus K forces the module to have only loops and intra-modular connections, causing the net to behave as an ordered one, even when connectivity is high. As we can see, the Attractors standard deviations for K_{TOT}=2 and K_{TOT}=3 are very big. This is because when there are only auto-interactions, almost every initial condition is an

N	K	M	L	T	K_{TOT}	A	Le	%SIA	μ	κ	σA	σL	$\sigma\%SIA$
1	0.5	20	0.5	20	1	31.67	2	0.0068	0.05	0.5	51.596	0.870	0.0122
1	1	20	1	40	2	197.17	3	0.0591	0.05	0.5	182.155	1.615	0.0600
1	1.5	20	1.5	60	3	119.84	4	0.0416	0.05	0.5	124.516	2.668	0.0446
1	2	20	2	80	4	59.44	5	0.0243	0.05	0.5	69.834	4.101	0.0281

Table 4.13: Statistics results for case 5: $N = 1 \quad L = K, \quad M = 20$

attractor. This could be interpreted as having few very long attractors and many small ones, near the phase transition.

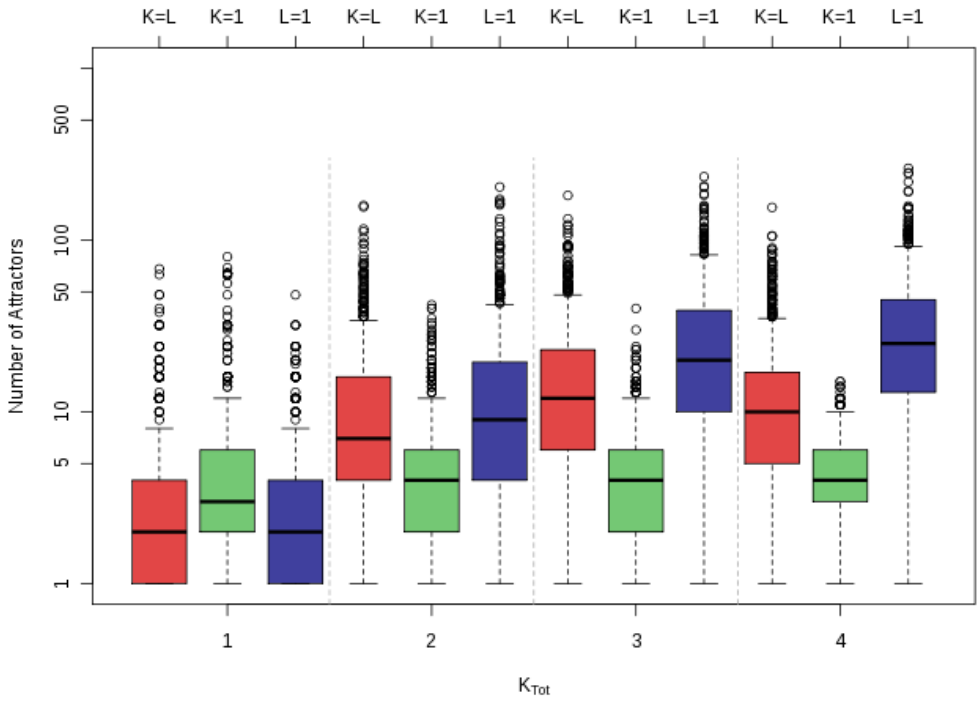

Figure 4.5: Number of Attractors vs. K_{TOT} for κ-exploration cases 1, 2 & 3 ($K = L$, $K = 1$ and $L = 1$, respectively).

In Figure 4.5 we can see[10] that only for $K_{\text{TOT}} = 1$ the inter-modular case (K=1, case 2) there are more attractors. As the *meta*-net connectivity T grows, the intra-modular case (L=1, case 3) becomes the one with more attractors, followed by the balanced case (case 1, K=L). The second case basically remains the same for all K_{TOT}.

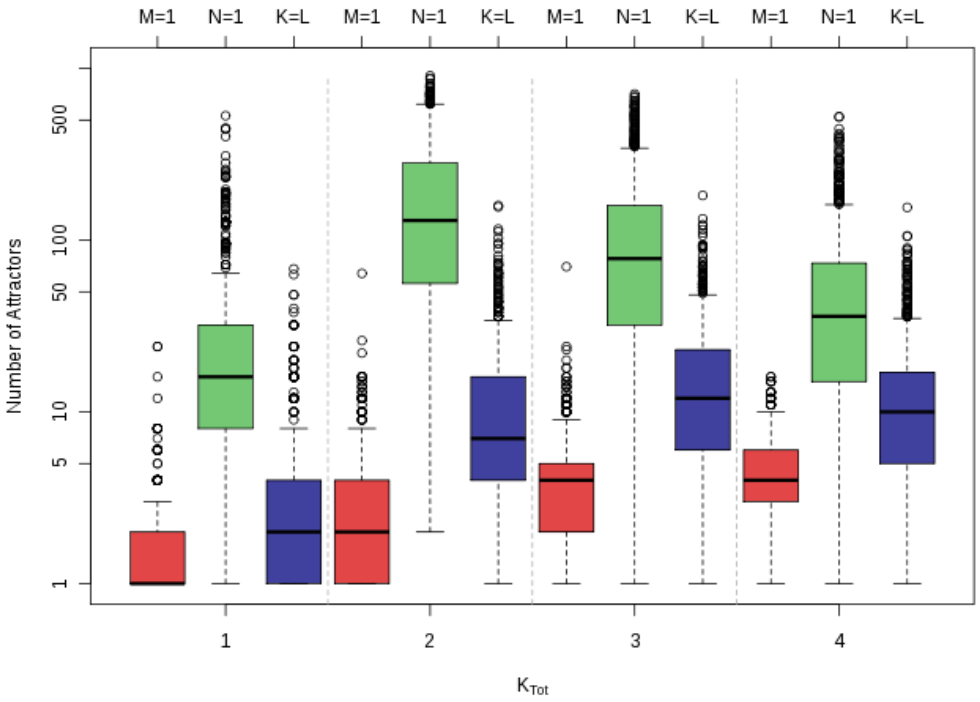

Figure 4.6: Number of Attractors vs. K_{TOT} for μ-exploration cases 4, 5 & 1 ($M = 1$, $N = 1$ and $K = L$, respectively).

[10] A detailed explanation of the way in which plots should be interpreted is available in appendix B.3.

Figure 4.6 shows that networks from case 5 ($N = 1, L = K$ and $M = 20$) tend to have more attractors. In these nets, it is more likely to be in an attractor than in other type of nets. As $N = 1$, there are few inter-modular links, therefore with low K values (as was the case) it is highly probable that the initial state is already an attractor.

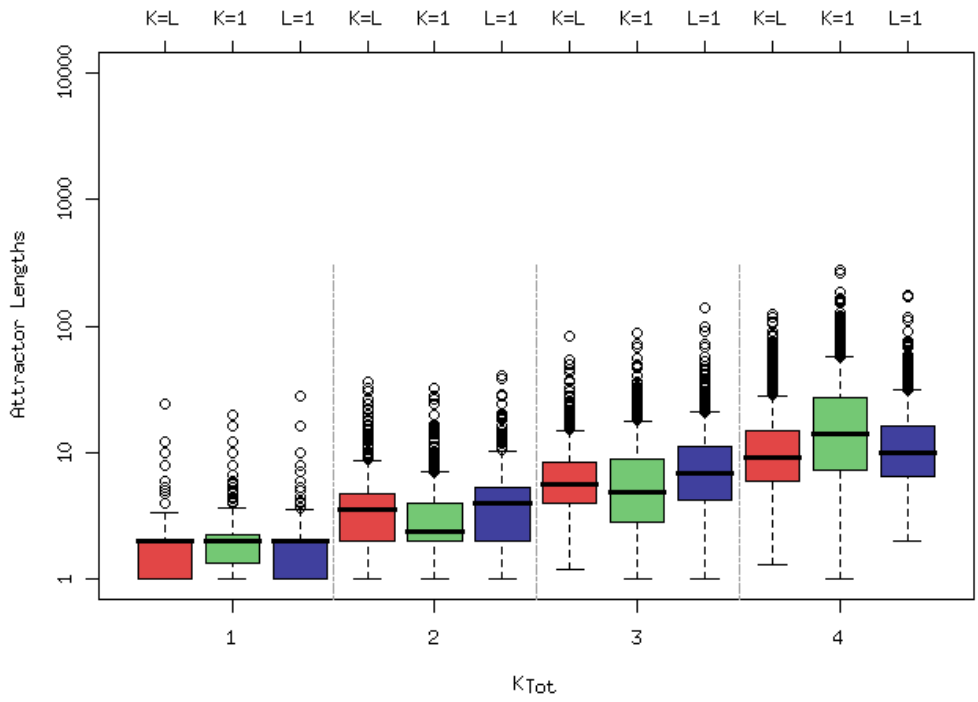

Figure 4.7: Average Length size for cases 1,2 & 3 ($K = L$, $K = 1$ and $L = 1$, respectively).

The average length size for the κ-exploration cases, shown in Figure 4.7, display a regular increment as K_{TOT} grows. All cases have more or less the same length size until K_{TOT}= 4, where the inter-modular case (K=1) separates away from the two other cases. We can see that attractor length is roughly independent of κ, although dependent on K_{TOT}.

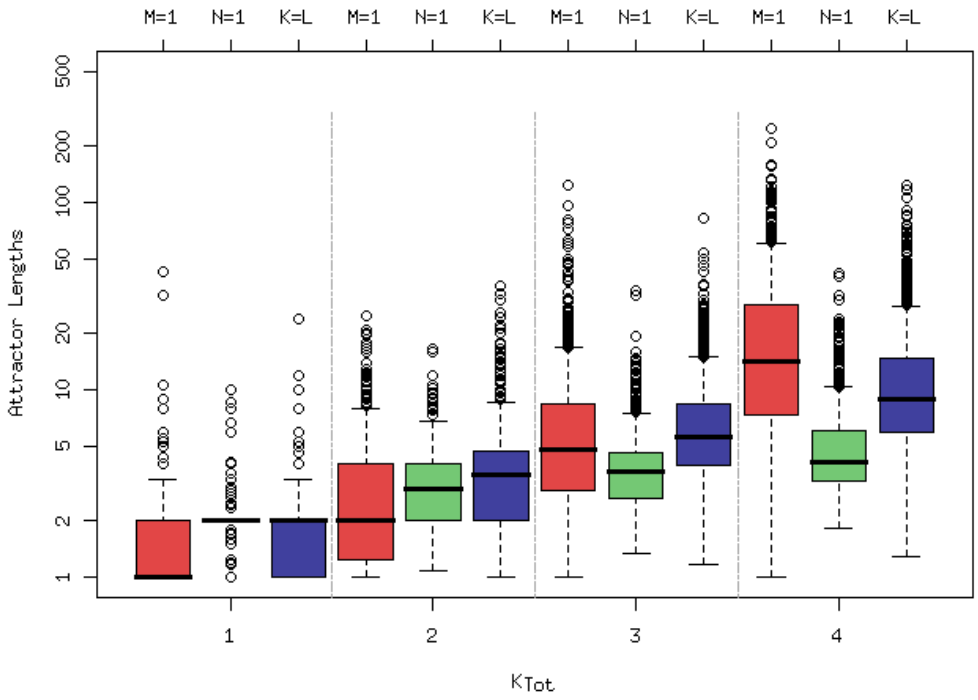

Figure 4.8: Average Length size for cases 4,5 & 1 ($M = 1$, $N = 1$ and $K = L$, respectively).

While comparing case 1 ($K = L$ and $\frac{N}{M} \rightarrow 1$) on Figure 4.8 with the other two cases, we see that the average length size is always higher or equal to the others. Here the auto-links make the lengths fairly short, appearing to remain the same for all K_{TOT}. For the fourth case (typical RBN) we see that for K_{TOT}= 4, its value is even higher than that of case 1 (rightmost). In case 5 ($N = 1$), lengths increase, although not as much as the other two cases. This implies that μ does affect attractor length.

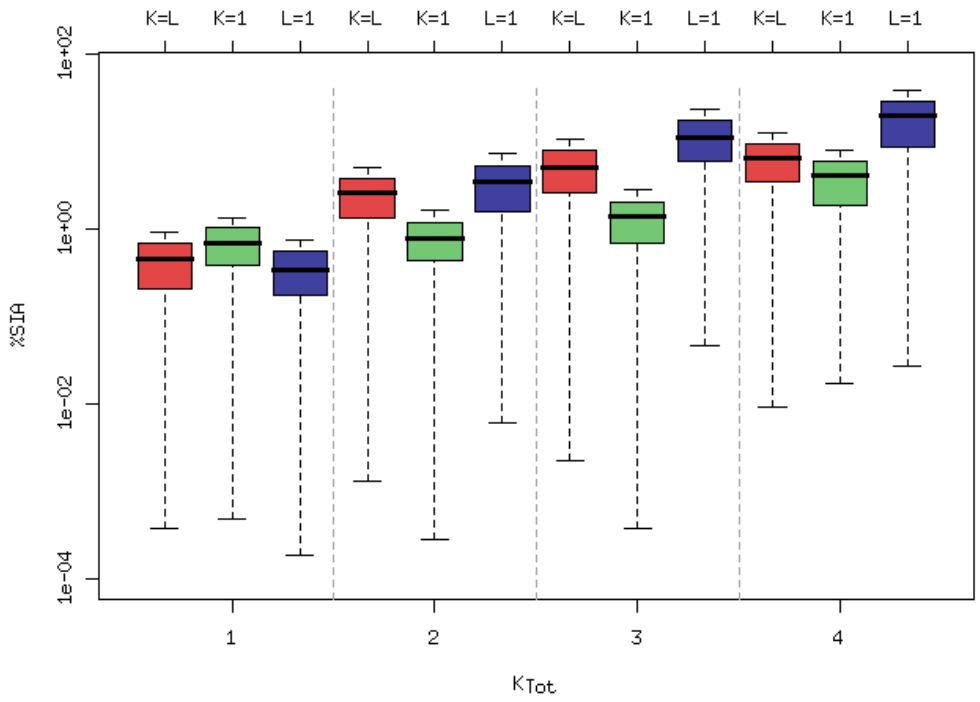

Figure 4.9: Percentage of States in Attractors for various values of K_{TOT} in the κ-exploration cases 1,2 &3 ($K = L$, $K = 1$ and $L = 1$, respectively).

For both graphics of %SIA (Figures 4.9 and 4.10) the scale is logarithmic for the y-axis. This was chosen because of the large number of states that exist in the state-space. In particular for the first figure, we see that %SIA for K_{TOT}= 1 is not even 1%. For other values of K_{TOT} the inter-modular case (green) remains very close to 1 percent, while the balanced case (red) and the inter-modular case (blue) rise about a hundredth of that 1%. We could say that for all except K_{TOT}= 1 the intra-modular cases are the most stable ones.

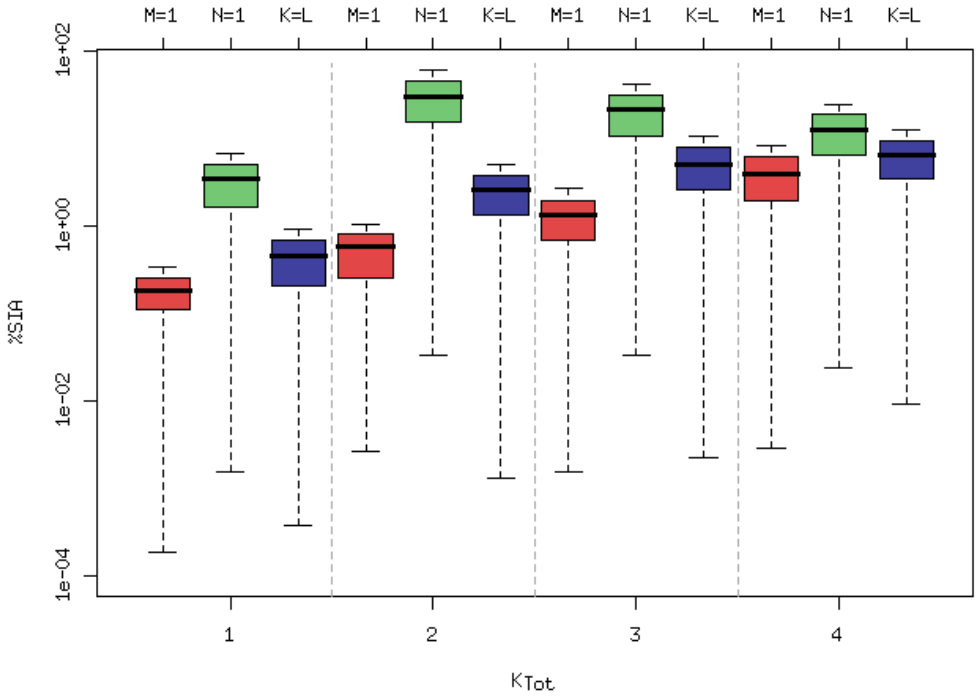

Figure 4.10: %SIA in the μ-exploration cases 4,5 & 1 ($M = 1$, $N = 1$ and $K = L$, respectively).

Here case 5 has the obvious result, because many of the links within the networks are loops, and loops create attractors, therefore the probability of a state being within an attractor is higher. $K_{\text{TOT}}= 2$ and $K_{\text{TOT}}= 3$ show how modularity increases the stability of the network by having more states in attractors. The other cases (typical RBN and balanced RBN) show a good percentage of states in attractors, being the balanced the one that has more between both. Nonetheless, the lower quartiles for all K_{TOT}s in cases 4 and 1 (red and blue, respectively) are very low, showing that few of the nets actually had a more unstable configuration.

Chapter 5

Discussion & Conclusions

5.1 Discussion

In the previous chapter we showed how modularity acts on random Boolean networks and its direct effect on the net's criticality. It was originally showed that for classical RBNs, dynamics reach a critical level when the parameter K is equal to 2; fact that was originally proved by Derrida & Pomeau [Derrida and Pomeau, 1986].

As has been found by experiments that reach several areas of research, modularity appears to be ubiquitous [Han et al., 2004, Gavin et al., 2002, Uetz et al., 2000, Schilling, 2000, Fodor, 1983]. Despite the difficulty that the *precise* measure of modularity might bring to us, it is undeniable that natural systems often have it. The main interest of this work was to determine how modularity affects the criticality of RBNs, and —in a second layer— find out if a system is more adaptable when being modular.

We showed that modularity influences more on the information transference than the typical less realistic supposition of their regular topology. In order not to restrict modularity to only some parameters, we have been varying modularity with both μ and κ, that is the relation of N and M and the relation of K and L. Experiments on μ and κ they both support the hypothesis that modularity —in both *measures of modularity*— influences criticality.

We found that mRBNs are less sensitive to initial conditions. This was mainly seen when modular networks (specifically cases 1: $K = L$ and μ fixed and 3: $L = 1$ and μ fixed) were compared with traditional RBNs (case 4: $M = 1$, $N = 20$, $L = 0$ and $K_{\text{TOT}} = K$), where the contrast of the presence of modules shows mRBNs have more stability. But with more subtlety, the presence of weak links made mRBNs even more stable than those that had strong links instead (cases 2 vs. 3), i.e. those with higher connectivity between modules (refer to tables 4.1 - 4.5). Just as seen earlier in 3.1 Weak Links, there is a straight relation between our connectivity variables' proportion (K and L, i.e. λ), and the one between strong and weak links; these weak links unite modules and **create** the *meta*-net. Indeed, any damage mutation has less probability of being transferred from one module to another, when the amount of weak links (ℓ_w, that is $L \times M$) is less than that of strong links (ℓ_s, that is $K \times \mu$), in other terms $\ell_w < \ell_s$. This could seem obvious, because if there are less ways to propagate damage, the risk of "infecting" other modules is trivially less; but the next question is, even if this probability is low, what happens when this mutation spreads along the *meta*-net? A real stable system should be able to cope with (up to certain level of) mutations that run through the net. And this is precisely what case 3 ($L = 1$ with μ fixed) does. This is due to a low number of inter-modular (weak) links, with a larger number of intra-modular (strong) links, which constrain mutations within a module. Keeping a mutation (damage) inside a module, allows the network's dynamics to

adapt to it and modify its function as little as possible. Once a module has coped with damage, a mutation can be transmitted to other modules. This could be safely done after a threshold of time, because inter-modular links do not abound, thus letting other modules to (possibly) cope with damage.

Case 1 ($K = L$ and $N \to M$, balanced intra/inter modular links) shows another interesting thing. There is a big leap between $K_{\text{TOT}} = 3$ and $K_{\text{TOT}} = 4$. As shown in table 4.1, in these mRBNs we know that the critical region broadened from values $K \approx 2.5$ and $K \approx 3$, that is, dynamics within the chaotic phase became less sensitive to initial conditions, bringing them closer to the critical regime than without modularity (case 4). Two new cases were added for table 4.1 to show the properties for both (STIC & Statistics) experiments for case 1 ($K = L$, balanced intra/inter modular links). By this addition, we can observe how the critical region expands, and it is easily compared to traditional RBNs (case 4: $M = 1$, $N = 20$, $L = 0$ and $K_{\text{TOT}} = K$).

Figure 5.1 depicts $\overline{\Delta H}$ for case 1 ($K = L$) and case 4 (typical RBN). As we can see, after $K = 2$, the red curve (case 1) stays below the green curve (case 4, typical RBN), then for $K = 2.5$ the typical RBN has larger value than the balanced case, they finally almost join at $K = 3$ and from that point on, case 4 simply goes too chaotic. This is due to modularity, because if we recall there is a balance of inter-modular and intra-modular links for case 1.

Now we'll compare the same typical RBN (case 4) with the mRBN which contains few weak links and therefore avoids damage spread. As we can see in Figure 5.2 although case 1 ($K = L$) was better than its balanced modular counterpart, here case 3 ($L = 1$ with μ fixed) is even better. We can see that for $K = 2.5$ the green curve is separated enough from the red one. For next values, case 3 has less increment than case 1 which

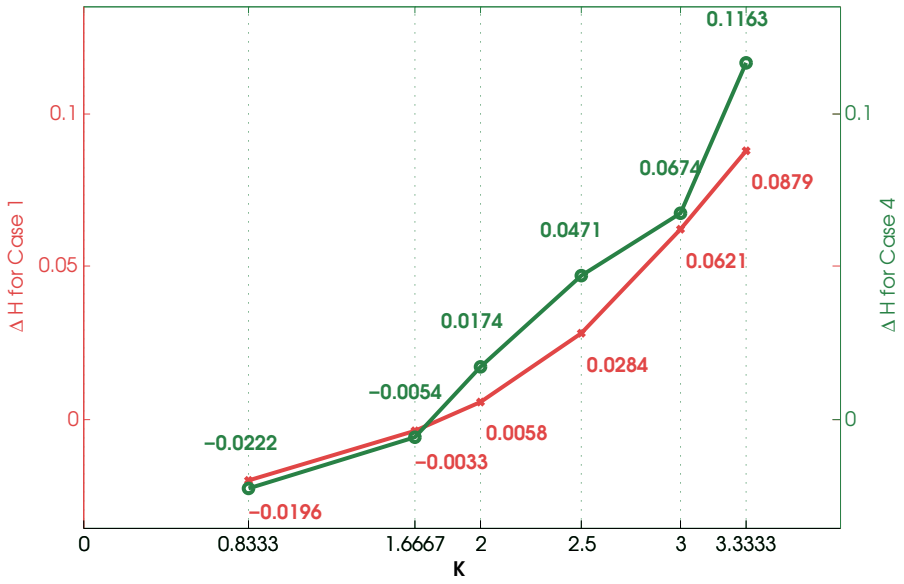

Figure 5.1: Comparison of the STIC ΔH values for cases 1 and 4 ($K = L$ and $M = 1$, $N = 20$, $L = 0$ and $K_{\text{TOT}} = K$ respectively).

shows being more sensitive to damage. This fact supports the hypothesis that modularity with low L values —that is, weak links— copes better with damage than the balanced one, which —while being way better than typical RBNs— spreads more damage because it has a higher connectivity.

Another comparison that supports our theory, is the number of attractors the data of the *Statistics* experiment showed. We will first discuss the "obvious" case 5 ($N = 1$, $L = K$ and $M = 20$), because by definition it contains isolated modules which in fact could be seen as single nodes, thus links will most surely be auto-links. Given this, damage cannot easily spread, but so cannot information flow, therefore it would take a longer time to "deal" with any noisy signal from the exterior (environment) and even from within the *meta*-net.

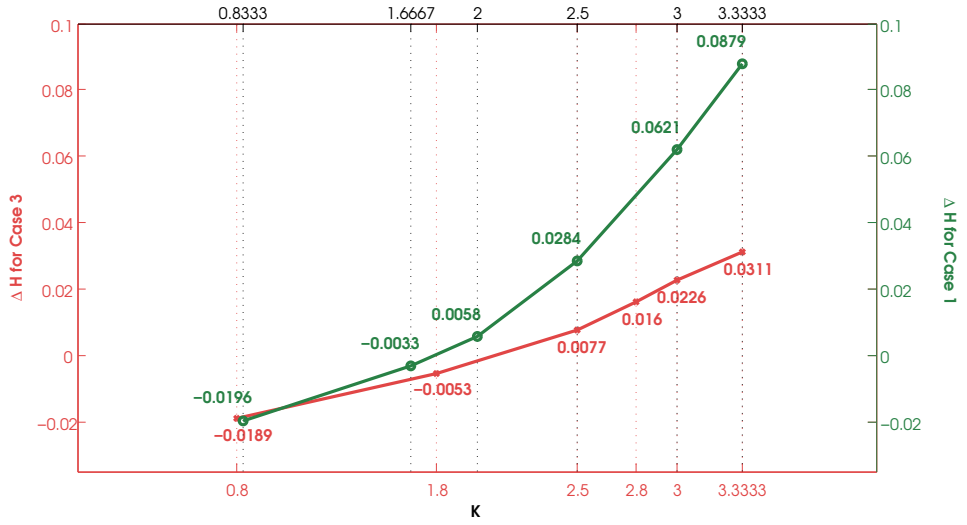

Figure 5.2: Comparison of the STIC ΔH values for cases 1 and 4 ($K = L$ and $M = 1$, $N = 20$, $L = 0$ and $K_{\text{TOT}} = K$ respectively).

The following two figures display the statistics results for both cases 1 and 4 respectively.

The number of attractors of the balanced case in Figure 5.3 shows that these kind of networks have their maximum when K_{TOT}=3, and there is a sudden decrease again when K_{TOT}=4.

Compared side-by-side as in Figure 5.5 we can see that the number of attractors for case 4 grows almost linearly and very slow. The most distant value occurs at K_{TOT}=1, and the average attractor count is $\approx$ 3.4. The other case compared is case 1 whose average attractor count is $\approx$ 12.21 As we mentioned before, the critical region is stretched out by modularity when $2.5 \lessapprox K_{\text{TOT}} \lessapprox 3.5$, which is evident while compared with the typical RBN which reacts almost equally for all cases.

Comparing cases 3 and 4 ($L = 1$ and μ fixed and $M = 1$, $N = 20$, $L = 0$ and $K_{\text{TOT}} = K$,

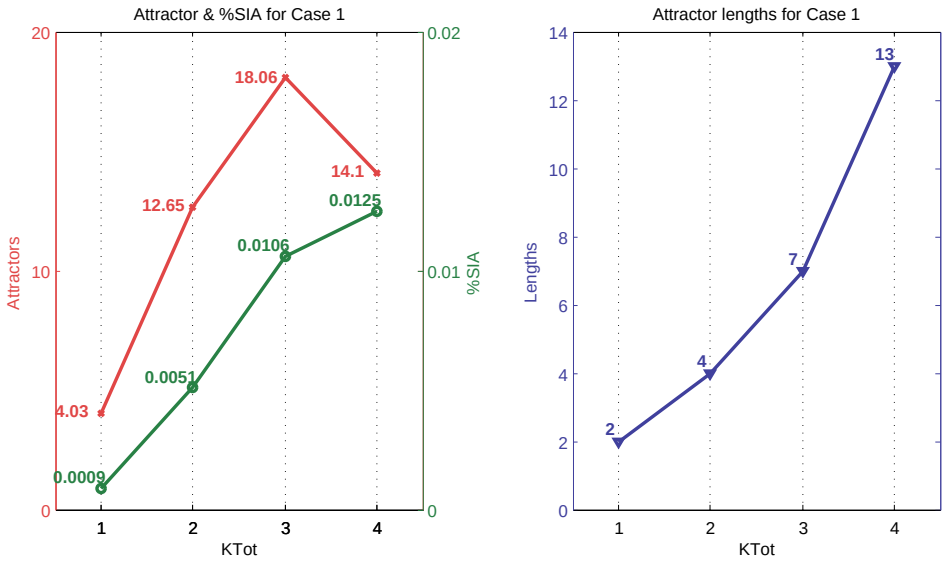

Figure 5.3: Statistics experiments results for case 1 ($K = L$).

respectively) shows the payoff of being *more* modular. The typical RBN (case 4) seems to be a horizontal line with respect of the modular one (case 3 in red).

Looking into the number of attractors shown in Tables 4.13 and 4.11, and comparing any to the rest of the tables, we can clearly see a burst. Given the nature of case 5 the high number of attractors is normal, because many of those attractors are in fact loops. We would have expected something similar comparing cases 1 and 3. Of course not the same amount of attractors, but indeed higher than other cases. We can see the effects of modularity on these two cases in Figure 5.7. The mentioned burst in the number of attractors is due to the multiple combinations of local interactions. That is, there are some attractors local only to the module and some others over the *meta*-net, thus when combining and counting them, they appear to be much more.

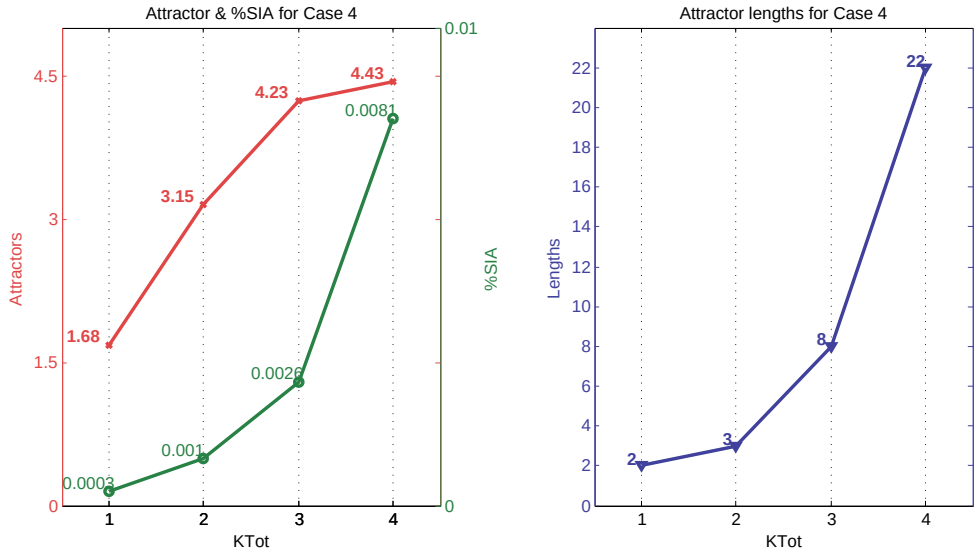

Figure 5.4: Statistics experiments results for case 4 ($M = 1$, $N = 20$, $L = 0$ and $K_{\text{TOT}} = K$).

The behaviour of cases 1 and 3 should be seen together too. For the first two values of K_{TOT} they are essentially equal, but as the critical region is reached, case 3 has substantially more attractors than case 1 but for K_{TOT}=4, case 3 has more than twice the attractors as case 1 has.

Let us remember that case 1 has the restriction $K = L$. That is, there is an exact balance between inter-modular links (edges between modules) and intra-modular links (edges between nodes). Some of the attractors found for case 3 possibly involve sub-attractors of each module. These attractors use both strong and weak links, so the net's dynamic flows along strong links and with lower probability through weak links, giving the net various additional attractors compared to its balanced sister from case 1.

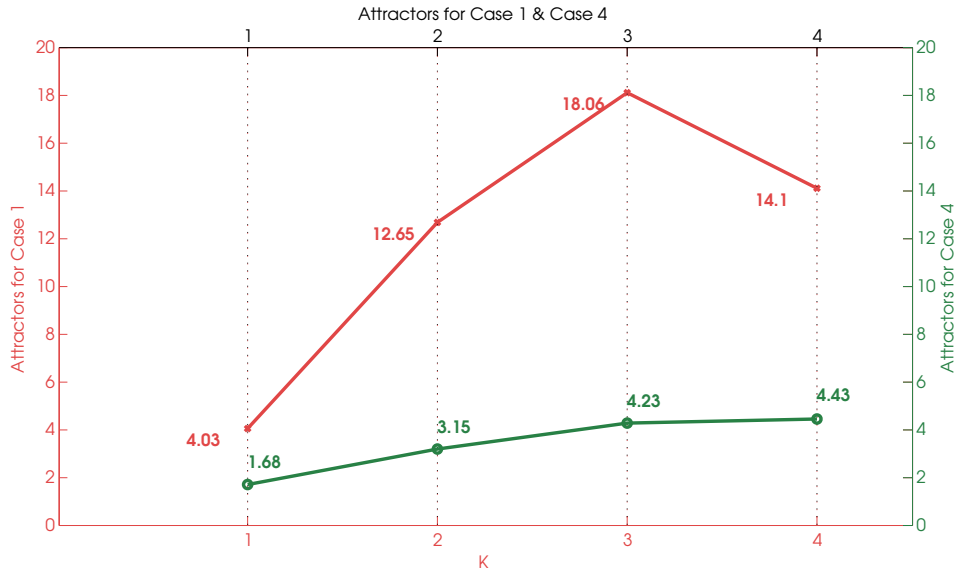

Figure 5.5: Comparison of the statistics results for cases 1 and 4 ($K = L$ and $M = 1$, $N = 20$, $L = 0$ and $K_{\text{TOT}} = K$ respectively).

5.2 Future Work

At a small seminar which took place in Spring 2010 at C3 headquarters in Mexico City, my supervisor Carlos Gershenson and I talked with Max Aldana, and the notion of parallelising the algorithms that analyse RBNs (and mRBNs) was asked; Max wanted to do this to be able to evolve and analyse big nets (N=1000).

This is certainly an important topic to focus on, because making the algorithms parallel not only requires a full new (re)design of the software, but allows us to acquire more significant statistical data for the analyses and to deepen the way properties of RBNs are sought. It was mentioned in section 4.2.2 that one of the main limiting issues encountered was computational power. Real GRNs have hundreds —when small—, if not thousands of genes, therefore this work's approach would appear rather small compared to what Nature

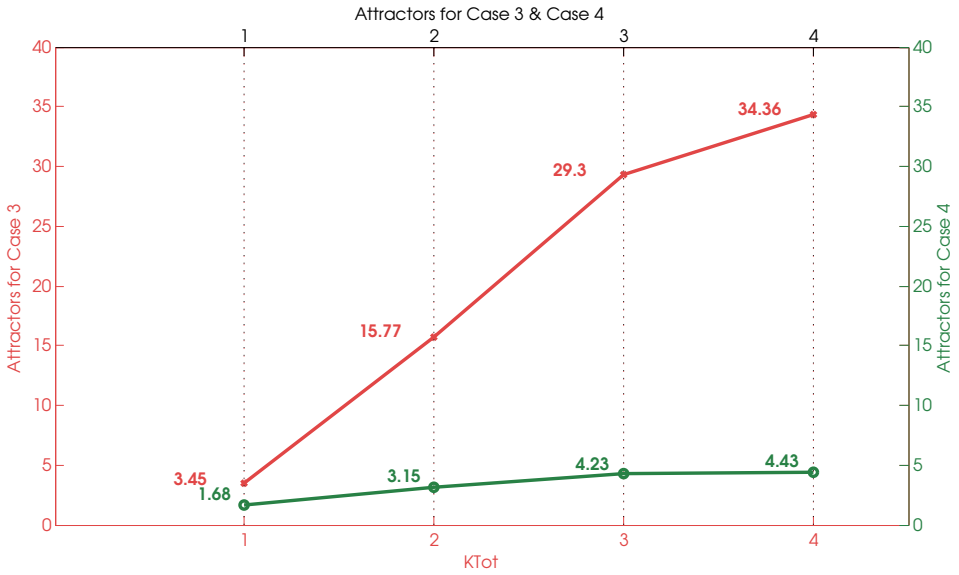

Figure 5.6: Comparison of the statistics results for cases 3 and 4 ($L = 1$ and μ fixed and $M = 1$, $N = 20$, $L = 0$ and $K_{\text{TOT}} = K$, respectively).

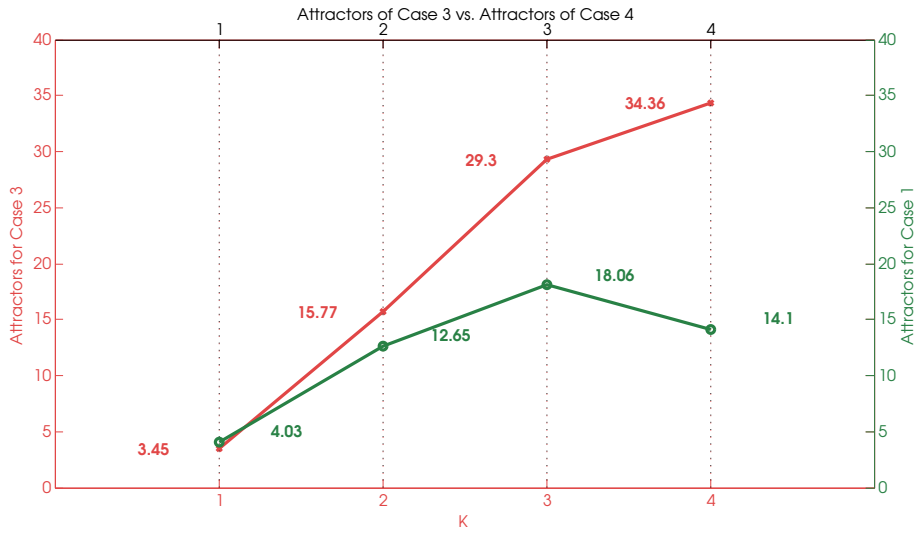

Figure 5.7: Comparison of the statistics results for cases 3 (in red) and 1 (in green) ($L = 1$ and μ fixed and $K = L$, respectively).

really exposes us to, although the results hereby usefully point towards a specific direction.

Parallelising would imply the embracement of MPI (Message Passing Interface) for a low-level programming language like C/C++, although some libraries exist that support Java [Carpenter et al., 2003], which imply that the RBN-Lab could be *extended* to be parallel. It would require an enormous effort, and probably the assistance of a group of experts on concurrent and parallel programming and algorithms, and of course a multi-processor computer or a cluster.

Another approach here would be to compile statistical data focused on noise. Barbieri, Serra and company studied the way noise influences RBNs' dynamics [Barbieri et al., 2009] and found out that small amounts of noise[1] that alters the nets, make the dynamics jump from one attractor to another. When neglecting infrequent noisy signals, patterns are found that arise from initial patterns. Finally, making the net grow in the number of nodes, increases "for free" the latter phenomenon, creating an interesting sequential specialisation of the initial undifferentiated patterns. It would be interesting to mix the method applied in [Barbieri et al., 2009] to mRBNs and observe how dynamics respond to the noisy fluctuations and determine how the patterns change, given that modularity is present in the RBN.

Lizier, Prokopenko & Zomaya in their 2008 paper [Lizier et al., 2008] investigated the phase transition between order and chaos in RBNs behaviour from a *distributed computation* taken place in their nodes; they characterise this computation in terms of *information dynamics* taken from information theory. Three measures were made:

- Information storage

[1] The focus on a "small" noise fluctuation which is driven by single bit flips during execution.

- Information transfer
- Information modification

Not deepening too much into the way in which they obtained the results, they basically showed that entropy increases with the average connectivity $\overline{K}$. Somewhere in the range $\overline{K} = (2, 2.5)$ of the average connectivity, the active information reaches its maximum and later decays to values near zero. After $\overline{K} = 1$ and before $\overline{K} = 2$ this value is always above zero and growing at the same rate as entropy does. Just around the point where the active entropy reaches its maximum value, the entropy rate begins to grow exponentially, matching entropy near $\overline{K} = 3$. Using these measures they conclude that the ordered phase is dominated by information stored in the past of the node about its future state: information storage; and the chaotic phase is dominated by information transfer, that is, information from the input of other nodes which dictates the current node's next state. The critical regime shows a balance between storage and transfer. Near the transition phase there is a large amount of trivial information modifications, leading to the implication that GRNs have evolved to maximise the coherent computation.

Applying this framework to mRBNs would probably show interesting results, and we could be able to determine if the critical regime loosens or tightens in the same range of K_{TOT}; it could also be shifted up to values of higher K_{TOT} near to 3. Thus this too would be an interesting perspective to research.

Max Aldana, in his 2003 paper *Boolean dynamics of networks with scale-free topology* [Aldana, 2003] studied the dynamics on networks with Scale-Free topology. He showed analytically the existence of the phase transition between ordered and chaotic behaviour, which is in fact determined by the network's scale-free exponent. The phase transition was

found within the interval (2, 2.5) of the net's scale-free exponent. A question we could ask now is, what would happen if we made those networks modular? Perhaps the above interval would be wider. Therefore it is also important to study how modularity behaves in this topology.

As Bilke and Sjunnesson [Bilke and Sjunnesson, 2001] and later Socolar [Socolar and Kauffman, 2003] suggest, there is a way in which the *irrelevant* nodes can be discarded. Irrelevant nodes are those which do not alter their function no matter what input they receive, thus removing them does not alter the dynamics. The steps applied to remove irrelevant nodes are [Socolar and Kauffman, 2003]: a)Identify "fixed" nodes whose outputs are entirely independent of their inputs. b)With an iterative procedure frozen nodes are identified. These frozen nodes become frozen because their outputs depend of the inputs of (already) frozen nodes. Frozen nodes are also called clamped nodes. c) A similar procedure is used to prune nodes with no relevant outputs, which are leafs in the tree. By removing irrelevant nodes from the network, it is easier to determine the exact number of attractors, the attractors average length and the number of nodes which truly participate in the attractors dynamics, thus RBNs with many more[2] nodes could be analysed, also for mRBNs. What is important to point out, is what for Socolar found for critical networks. The average number of fluctuating nodes grows like $N^{2/3}$, as opposed to previous results which revealed a $\sqrt{N}$ growth. They found that the median number of attractors grows faster than linearly at least for N up to 1200. The author believes this difference in the results are due to not considering largely enough N. Therefore this procedure could be used as a pre-processing procedure for both kinds of experiments STIC and Statistics and have simulations take less time for mRBNs with thousands of nodes, instead. It would be interesting to see how significant the value of N becomes and how the number of attractors varies.

[2]Than the number studied in this work.

Finally, the wide range of rubrics that Carlos Gershenson exposes in *Guiding the Self-organization of Random Boolean Networks* [Gershenson, 2010] to guide RBNs towards self-organisation. He reviews eight methods to do so, and in particular guiding RBNs towards the critical dynamical regime. As he states, we are interested in understanding how the interaction among nodes affect the whole net's properties and dynamics, thus this change of scale —from the lower scale (nodes) to the higher scale (whole net)— favours the description and allows us to better guide RBNs towards self-organisation. The eight methods are p modification, K modification, bias towards canalising functions, silencing nodes, altering the topology either changing the link distribution or the link regularity, affecting modularity, adding redundancy and applying degeneracy.

5.3 Conclusions

(M)RBNs are our model of GRNs. It is our way to explain how modularity acts on the criticality of mRBNs which are in fact a good *simple* representation of GRNs. We have shown that networks are a very good representation for "everyday problems" such as the ones originally proposed by Euler; but they are also good ways of representing less evident "everyday problems", as how genes are regulated in living systems.

Defending both ideas, the idea that it is believed that life and computation take place within the critical regime, and that modularity is an ubiquitous property of natural (and some artificial) systems, we showed an idea of what happens when these two basic concepts permeate in random Boolean networks. Modularity in this sense acts as a buffer between extreme change (chaos) and extreme steadiness (order); if we saw each regime as having a probability of occurrence p_o, this buffer lowers the chaotic and order's p_o, thereby raising

the critical p_o. This is what we showed in past chapters: the critical region widens when the network is modular, making the net more robust. So if the same applies to GRNs, we could argue that in natural systems modularity has *happened* in order to allow information to be transferred along time, being safely transmitted and computed in the critical region. Modularity exists basically because solutions are re-used, whether they are efficient, optimal or safe, but in long terms it is a good component for evolution.

We also exposed how both types of RBNs, the modular and the "typical" ones react to small changes in initial conditions. The mRBNs clearly result less affected by these changes, making them —if not fully predictable— at least less chaotic. As some comparative results have shown, modular RBNs almost behave as robust as isolated modules themselves, reducing the damage spread along the whole *meta*-net.

The importance of weak links versus strong links is a relevant topic too. This is because they remarkably influence in both making the nets more robust and broadening the critical region. Chiefly in the statistics experiments 4.2.2 we exhibited how their presence influenced the net's properties. There also needs to be a balance between weak and strong links, otherwise the general dynamics of the system can flow towards a more ordered state (with no or few weak links) or towards a more chaotic state (with many weak links).

Complex systems are not easily understood and often, while trying to explain them, we run into new properties and unanswered questions. That is why scientists at a given time have to explore few aspects of them. It turns out that sometimes answering a question about a model, also answers (or approaches) some aspect of a "real world" system.

Timescale is one of the main issues for understanding many processes of nature. Our senses —and thus all our nervous system— have adapted to perceive events in a *meso*-timescale.

Events that take seconds, hours, days or months, a few decades at most. That is why so many natural events cannot be explained straight forward: we need time. Evolution is a process that takes place not only in a thousand- or million year scale, but also at a micro or nano second scale. In a way, the effects at a micro level also take place at a macro level.

I firmly agree that life is neither the result of randomness nor of pure chance, and it would be (almost) impossible to randomly arrange a large number of molecules and have a fern as a result, nor would it be likely that the outcome were a rock. Physical processes or entities could be seen as simple, because they don't serve any (apparent) purpose. A falling rock will fall if dropped from above two meters. It would not be arguable to say that the rock's purpose is to fall, because there is no (apparent) goal on falling and there is no way the rock can avoid falling. The rock is obliged to obey the law of gravity. Things without a purpose are usually not designed, that is, there was no "thinking" behind them. Certainly following a rule (or physical law) does not imply a design. In no way am I trying to say that physical entities (which actually follow those laws) are not complex, but they are there because that's where the laws of Nature have driven them to. In a way we could say that these physical things have life, a simple kind of life. Of course biological entities are not excluded from following the laws of Nature; but in a way, living organisms are always *fighting* against some natural or physical law. There is also computation at another contextual level [Hopfield, 1994]; in other words, living systems do more (than other systems, like artificial ones). Biological things will continue to change in order to adapt to those rules imposed by Nature. If we now drop a bird, we would see that way before it hits the ground, it would fly. The evident purpose here could be not to die. I would say that not to die is the bird's purpose within our timescale, but there is a purpose at a larger timescale not always too evident: life. Life as the result of an **apparent** design which always fights back to continue to exist. It is actually a blind design, it's adaptation

through trial and error.

One of the most complex systems we know of is the human mind. It is an entangled intangible system whose output can either be tangible or intangible. This complex system called mind is the result of many trials and many errors made by Nature that provides us with consciousness to admire what surrounds us: Life!

Bibliography

[Albert et al., 1999] Albert, R., Hawoong, J., and Barabási, A.-L. (1999). The diameter of the world-wide web. *Nature*, 401:130–131.

[Aldana, 2003] Aldana, M. (2003). Boolean dynamics of networks with scale-free topology. *Physica D*, 185:45–66.

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

[Andrecut, 2005] Andrecut, M. (2005). Mean field dynamics of random Boolean networks. *Journal of Statistical Mechanics*, P02003.

[Andrecut and Ali, 2001] Andrecut, M. and Ali, M. K. (2001). Chaos in a Simple Boolean Network. *International Journal of Modern Physics B*, 15:17–23.

[Barabási, 2002] Barabási, A.-L. (2002). *Linked: The New Science of Networks.* Basic Books, 1st edition.

[Barabási and Bonabeau, 2003] Barabási, A.-L. and Bonabeau, E. (2003). Scale-free networks. *Sci Am.*

[Barabási and Albert, 1999] Barabási, A.-L. and Albert, R. (1999). Emergence of scaling in random networks.

[Barbieri et al., 2009] Barbieri, A., Serra, R., Villani, M., Kauffman, S. A., and Colacci, A. (2009). On the dynamics of random Boolean networks subject to noise: Attractors, ergodic sets and cell types. *Journal of Theoretical Biology*, 265(2):185–193.

[Berlow, 1999] Berlow, E. (1999). Strong effects of weak interactions in ecological communities. *Nature*, 398(6725):330.

[Biggs et al., 1986] Biggs, N., Lloyd, E. K., and Wilson, R. J. (1986). *Graph Theory, 1736-1936*. Clarendon Press, New York, NY, USA.

[Bilke and Sjunnesson, 2001] Bilke, S. and Sjunnesson, F. (2001). Stability of the Kauffman model. *Physical Review E*, 65(1):016129+.

[Bunde, 1995] Bunde, A. (1995). *Fractals in Science*. Springer-Verlag.

[Callebaut and Rasskin-Gutman, 2005] Callebaut, W. and Rasskin-Gutman, D. (2005). *Modularity: Understanding the Development and Evolution of Natural Complex Systems (Vienna Series in Theoretical Biology)*. The MIT Press.

[Carpenter et al., 2003] Carpenter, B., Getov, V., Judd, G., Skjellum, A., and Fox, G. (2003). Mpj. http://www.hpjava.org/papers/MPJ-CPE/cpempi/cpempi.html.

[Chambers et al., 1983] Chambers, J. M., Cleveland, W. S., Kleiner, B., and Tukey, P. A. (1983). *Graphical Methods for Data Analysis*. Chapman and Hall, New York.

[Csermely, 2001] Csermely, P. (2001). Water and cellular folding processes. *Cell Mol Biol*, 47(5):791–800.

[Csermely, 2009] Csermely, P. (2009). Number 2. Springer-Verlag, Berlin Heidelberg, second edition.

[Cvitanović et al., 2010] Cvitanović, P., Artuso, R., Mainieri, R., Tanner, G., and Vattay, G. (2010). *Chaos: Classical and Quantum*. Niels Bohr Institute, Copenhagen. ChaosBook.org.

[Derrida and Pomeau, 1986] Derrida, B. and Pomeau, Y. (1986). Random networks of automata: A simple annealed approximation. *EPL (Europhysics Letters)*, 1(2):45–49.

[Di Paolo, 2001] Di Paolo, E. (2001). Rhythmic and non-rhythmic attractors in asynchronous random Boolean networks. *Biosystems*, 59(3):185–195.

[Douglas et al., 2001] Douglas, S., Zauner, S., Fraunholz, M., Beaton, M., Penny, S., Deng, L., Wu, X., Reith, M., Cavalier-Smith, T., and Maier, U. (2001). The highly reduced genome of an enslaved algal nucleus. *Nature*, 410(6832):1091–6.

[Fodor, 1983] Fodor, J. A. (1983). *The Modularity of Mind: an Essay on Faculty Psychology*. The MIT Press, Cambridge, MA.

[Gavin et al., 2002] Gavin, A.-C., Bosche, M., Krause, R., Grandi, P., Marzioch, M., Bauer, A., Schultz, J., and Rick, J. M. (2002). Functional organization of the yeast proteome by systematic analysis of protein complexes. *Nature*, 415(6868):141–147.

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

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

[Gershenson, 2008] Gershenson, C., editor (2008). *Complexity: 5 Questions*. Automatic Press / VIP.

[Gershenson, 2009] Gershenson, C. (2009). Enfrentando a la complejidad: Predecir vs. adaptar. http://uk.arxiv.org/abs/0905.4908.

[Gershenson, 2010] Gershenson, C. (2010). Guiding the self-organization of random Boolean networks. *ArXiv e-prints*.

[Giacobini et al., 2006] Giacobini, M., Tomassini, M., De Los Rios, P., and Pestelacci, E. (2006). Dynamics of scalefree semi-synchronous Boolean networks. In Rocha, L. M., Yaeger, L. S., Bedau, M. A., Floreano, D., Goldstone, R. L., and Vespignani, A., editors, *Artificial Life X, Proceedings of the Tenth International Conference on the Simulation and Synthesis of Living Systems.*, pages 1–7. MIT Press.

[Han et al., 2004] Han, J.-D. J., Bertin, N., Hao, T., Goldberg, D. S., Berriz, G. F., Zhang, L. V., Dupuy, D., Walhout, A. J. M., Cusick, M. E., Roth, F. P., and Vidal, M. (2004). Evidence for dynamically organized modularity in the yeast protein-protein interaction network. *Nature*, 430(6997):380.

[Harvey and Bossomaier, 1997] Harvey, I. and Bossomaier, T. (1997). Time out of joint: Attractors in asynchronous random Boolean networks. In *Proceedings of the Fourth European Conference on Artificial Life (ECAL97)*, pages 67–75. MIT Press.

[Ho et al., 2005] Ho, M., Hung, Y., and Jiang, I. (2005). Stochastic coupling of two random Boolean networks [rapid communication]. *Physics Letters A*, 344:36–42.

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

[Hung et al., 2006] Hung, Y.-C., Ho, M.-C., Lih, J.-S., and Jiang, I.-M. (2006). Chaos synchronization of two stochastically coupled random Boolean networks. *Physics Letters A*, 356(1):35 – 43.

[Kauffman, 1969] Kauffman, S. A. (1969). Metabolic stability and epigenesis in randomly constructed genetic nets. *Journal of Theoretical Biology*, 22(3):437–467.

[Képès, 2007] Képès, F., editor (2007). *Biological Networks*, volume 3 of *Complex Systems and Interdisciplinary Science*. World Scientific Publishing Co. Pte. Ltd., 57 Shelton Street, Covent Garden, London WC2H 9HE, first edition.

[Langton, 1990] Langton, C. (1990). Computation at the edge of chaos: Phase transitions and emergent computation. *Physica D: Nonlinear Phenomena*, 42(1-3):12–37.

[Lewin, 2004] Lewin, B. (2004). *Genes VIII*. Benjamin Cummings.

[Lizier et al., 2008] Lizier, J., Prokopenko, M., and Zomaya, A. (2008). The information dynamics of phase transitions in random Boolean networks. In Bullock, S., Noble, J., Watson, R., and Bedau, M. A., editors, *Artificial Life XI: Proceedings of the Eleventh International Conference on the Simulation and Synthesis of Living Systems*, pages 374–381. MIT Press, Cambridge, MA.

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

[Mitchell, 2009] Mitchell, M. (2009). *Complexity: A Guided Tour*. Oxford University Press, USA.

[Neuman, 2008] Neuman, Y. (2008). *Reviving the Living, Volume 6: Meaning Making in Living Systems*. Elsevier Science, San Diego, USA.

[Newman, 2001] Newman, M. E. J. (2001). The structure of scientific collaboration networks. *Proceedings of the National Academy of Sciences of the United States of America*, 98(2):404–409.

[Ott, 2001] Ott, L. (2001). *An introduction to statistical methods and data analysis*. Duxbury Pr., North Scituate, Massachussets, fifth edition.

[Paolo, 2004] Paolo, E. A. D. (2004). The circular topology of rhythm in asynchronous random Boolean networks. *BioSystems*, 73:141–152.

[Poblanno-Balp and Gershenson, 2010] Poblanno-Balp, R. and Gershenson, C. (2010). Modular random Boolean networks. In Fellermann, H., Dörr, M., Hanczyc, M. M., Laursen, L. L., Maurer, S., Merkle, D., Monnard, P.-A., Støy, K., and Rasmussen, S., editors, *Artificial Life XII Proceedings of the Twelfth International Conference on the Synthesis and Simulation of Living Systems*, pages 303–304. MIT Press, Odense, Denmark.

[Poblanno-Balp and Gershenson, 2011] Poblanno-Balp, R. and Gershenson, C. (2011). Modular random Boolean networks. *Artificial Life*.

[Schilling, 2000] Schilling, M. A. (2000). Toward a general modular systems theory and its application to interfirm product modularity. *The Academy of Management Review*, 25(2):312–334.

[Serra et al., 2008] Serra, R., Villani, M., Damiani, C., Graudenzi, A., and Colacci, A. (2008). The diffusion of perturbations in a model of coupled random Boolean networks. In Umeo, H., Morishita, S., Nishinari, K., Komatsuzaki, T., and Bandini, S., editors, *Cellular Automata*, volume 5191 of *Lecture Notes in Computer Science*, pages 315–322. Springer Berlin – Heidelberg.

[Simon and Ando, 1961] Simon, H. A. and Ando, A. (1961). Aggregation of variables in dynamic systems. *Econometrica*, 29:111–138.

[Socolar and Kauffman, 2003] Socolar, J. E. S. and Kauffman, S. A. (2003). Scaling in Ordered and Critical Random Boolean Networks. *Physical Review Letters*, 90(6).

[Solé and Fernández, 2003] Solé, R. V. and Fernández, P. (2003). Modularity for free in genome architecture?

[Solé et al., 2002] Solé, R. V., Salazar-Ciudad, I., and Garcia-Fernández, J. (2002). Common pattern formation, modularity and phase transitions in a gene network model of morphogenesis. *Physica A: Statistical Mechanics and its Applications*, 305(3-4):640 – 654.

[Uetz et al., 2000] Uetz, P., Giot, L., Cagney, G., Mansfield, T. A., Judson, R. S., Knight, J. R., Lockshon, D., Narayan, V., and Srinivasan, M. (2000). A comprehensive analysis of protein–protein interactions in saccharomyces cerevisiae. *Nature*, 403(6770):623–627.

[Villani et al., 2006] Villani, M., Serra, R., Kauffman, S. A., and Ingrami., P. (2006). *Coupled Random Boolean Network Forming an Artificial Tissue*, volume 4173 of *Lecture Notes in Computer Science*, pages 548–556. University of Michigan Press.

[von Dassow et al., 2000] von Dassow, G., Meir, E., Munro, E. M., and Odell, G. M. (2000). The segment polarity network is a robust developmental module. *Nature*, 406:188–192.

[von Dassow and Odell, 2002] von Dassow, G. and Odell, G. M. (2002). Design and constraints of the drosophila segment polarity module: Robust spatial patterning emerges from intertwined cell state switches. *Journal Of Experimental Zoology (Mol Dev Evol)*, 294:179–215.

[Wagner et al., 2007] Wagner, G. P., Pavlicev, M., and Cheverud, J. M. (2007). The road to modularity. *Nature reviews. Genetics*, 8(12):921–931.

[Watts and Strogatz, 1998] Watts, D. J. and Strogatz, S. H. (1998). Collective dynamics of /'small-world/' networks. *Nature*, 393(6684):440–442.

[Wiener, 1965] Wiener, N. (1965). *Cybernetics, Second Edition: or the Control and Communication in the Animal and the Machine*. The MIT Press.

[Wikipedia, 2010] Wikipedia, T. F. E. (2010). http://en.wikipedia.org/wiki/Cybernetics.

[Wuensch, 1998] Wuensch, A. (1998). Discrete dynamical networks and their attractor basins. volume 6.

[Wuensche, 1994] Wuensche, A. (1994). Complexity in one-d cellular automata: Gliders, basins of attraction and the z parameter. Working Papers 94-04-025, Santa Fe Institute.

[Érdi, 2008] Érdi, P., editor (2008). *Complexity Explained*. Springer.

Appendix A

Code

The source code for the RBNLab, executions and plots can be found at http://rbn.sourceforge.net/.

Appendix B

Miscellaneous

B.1 Two to the twenty-five-thousandth

The result of computing $2^{25,000}$ in R, was Inf. Both Maple and KCalc gave the actual result of computing $2^{25,000}$.
$2^{25,000} = 5.62200981577e + 7525$

B.2 Statistics experiments limitations

In order for this experiment to be statistical significant, many combinations of initial states should be tested. Initially the nets were not limited to 20 nodes; they were in fact as big as eighty or one hundred nodes. Although the simulations of the *Statistics* experiment began in a later period than those of the *Sensitivity to initial conditions* experiments, the high amount of processor time shortened both experiments. The basic idea of this experiment

is to try many different networks, run them 10,000 time steps and save temporarily their properties to gather statistical information. As mentioned earlier, there are $\left(\frac{2^{2^K}N!}{(N-K)!}\right)^N$ possible networks for given K and N, thus, using the first parameters $N = 80$ and, for instance $K = 2$, that yields $101,120$; multiplied by the $10,000$ time steps we have $1,011,200,000$ and multiplied by the number of initial states each net was tested with, which was $\approx 2^N$, we have a total of more than $1,222$ quintillions (in the large scale), number extremely large even for powerful computers.

B.3 Box plot and its components

In the box plot, the lower and upper quartiles of the data are shown by the top and bottom of the rectangle, and the median is displayed by a horizontal line segment within the rectangle (the median is $Q(0.5)$). Dashed lines extend from the end of the boxes to *adjacent values*, defined as follows. The interquartile range is first computed, $IQR = Q(0.75) - Q(0.25)$. The upper adjacent value is defined as *the largest observation which is less than or equal to the upper quartile plus* $1.5 \times IQR$. The lower adjacent value is defined to be *the smallest observation that is less than or equal to the lower quartile minus* $1.5 \times IQR$. If any value y_i falls outside the range of the two adjacent values, its value is plotted as a single point; these are called "outliers".

Percentile. The **pth percentile** $Q(p)$ of a set of n measurements arranged in order of magnitude is that value that has at most p% of the measurements below it and at most (100 – p)% above it [Ott, 2001].

As stated in [Chambers et al., 1983], the box plot "gives a quick impression of certain prominent features of the distribution", for instance.

1. The median is shows the centre of the distribution.

2. The spread of the bulk of the data (the central 50%) is seen as the length of the box.

3. The lengths of the dashed lines relative to the box show how stretched the tails of the distribution are.

4. The outside values give a the opportunity to consider observations that seem unusual or implausibly large or small. Outside values are not necessarily outliers, but any outliers will appear as outside values.

Box plots also allow assessment of symmetry. If the distribution is symmetric, then the box plot is symmetric about the median: the median cuts the box in two exact halves, the upper and lower dashed lines are the same length and the outside values of top and bottom are equal in number and symmetrically placed.