



HAL
open science

On the dynamics of two particular classes of Boolean automata networks: Boolean automata circuits and OR networks

Mathilde Noual

► **To cite this version:**

Mathilde Noual. On the dynamics of two particular classes of Boolean automata networks: Boolean automata circuits and OR networks. 2009. hal-00713371

HAL Id: hal-00713371

<https://hal.science/hal-00713371v1>

Submitted on 30 Jun 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

**On the dynamics of two particular classes of Boolean
automata networks: Boolean automata circuits and OR
networks**

Mathilde Noual

Research Report February-May 2009
Report of an internship carried out between February and June 2009 at the
IXXI institute of Lyon under the supervision of Jacques Demongeot
(TIMC-IMAG of Grenoble) and at the CMM laboratory of Santiago de Chile
under the supervision of Eric Goles.

LONG VERSION WITH PROOFS

Abstract

The work presented here is set in line with fields of theoretical computer science and biology that study Boolean automata networks frequently seen as models of genetic regulation networks. In the context of biological regulation, former studies have highlighted the importance of circuits on the asymptotic dynamical behaviour of regulation networks. This is why we first chose to concentrate on networks whose underlying interaction graphs are circuits, that is, Boolean automata circuits. Here, we examine the dynamical behaviour of these networks in the case of a synchronous update schedule of their automata as well as in that of more general update schedules such as sequential or block sequential update schedules. Next, driven by the will to develop our understanding of networks with arbitrary underlying structures, we focus on OR networks and give some properties of their dynamics in an attempt to fully classify these networks according to their asymptotic dynamical behaviour considering all of their synchronous, sequential and block sequential update schedules.

Keywords. Discrete dynamical systems, Boolean regulatory network, positive and negative circuit, asymptotic behaviour, attractor, update schedule, OR network.

Contents

1	Introduction	2
I	Boolean automata circuits	4
2	Definitions, notations and preliminary results	4
3	Positive circuits	8
4	Negative circuits	10
5	Related problems	13
5.1	Binary shift register sequences	14
5.2	Binary necklaces and Lyndon words	15
5.3	Cycles in a digraph under $x^2 \bmod q$	16
6	Synchronous, sequential and block sequential update schedules	17
II	OR networks	24
7	Conclusion	32

1 Introduction

The theme of the research work I am about to present is set in the general framework of complex dynamical systems, and more precisely, that of regulation networks modeled by means of discrete mathematical tools. Since Kauffman [12] and Thomas [17] introduced the first models of genetic regulation networks at the end of the 1960's, many other studies based on the same or different formalisms were carried out in this context. One of the main motivations of these studies was to better understand those emergent dynamical behaviours that networks display and that cannot be explained or predicted by a simple analysis of the local interactions existing between the components of the networks.

Following the lead of some of the authors of these works, we decided to focus on *Boolean automata networks*, and more specifically on *threshold* Boolean automata networks and their dynamics. Informally, a **Boolean automata network** is defined by an interaction graph $G = (V, A)$ and a set of local transition functions f_i , one for each automaton i of the network, that is, one for each node $i \in V$. Every automaton or node is given a Boolean state that may change over time. The f_i functions are Boolean functions that allow the automata i to compute their Boolean states at time $t + 1$ knowing the states at time t ¹ of their incoming neighbours² in G . The arcs of A thus represent the dependencies there exists between the states of their extremities. **Threshold Boolean automata networks** are networks that were initially introduced by McCulloch and Pitts in [13] in order to represent neural networks formally. Their underlying interaction graph $G = (V, A)$ has weights $w_{i,j} \in \mathbb{R}$ on each arc $(i, j) \in A$, and their nodes $i \in V$ have thresholds $\theta_i \in \mathbb{R}$. Their local transition functions are of the form $f_i(x) = H(\sum_{j \in \mathcal{N}^-(i)} w_{j,i} \cdot x_j - \theta_i)$ where H is the Heaviside function ($H(x) = 0$ if $x < 0$ and $H(x) = 1$ if $x \geq 0$).

Amongst the features of a regulation network that may significantly impact on its dynamical behaviour, is the update schedule of its automata, that is the order with which the automata compute their new state within each time step. Here we make two hypotheses on the update schedules of networks we will consider : (i) the update schedule is deterministic and invariant with time, (ii) within every time step, each node is updated exactly once. Now, to construct systematically an understanding of networks dynamics considering all such update schedules, one can choose two different approaches : either simplify the underlying structure of the networks or simplify the transition function of the network, that is the way each automata computes its new state according to that of the automata that influence it. This is why, during this internship, I chose to study two types of networks : Boolean automata circuits, that is, networks whose underlying dependency or interaction graph is a circuit and *OR* networks, *i.e.*, networks whose transition functions are the

¹To be meticulous, I should add “provided the automata are updated synchronously” but the point here is only to give a general idea of what are Boolean automata networks. Details concerning update schedules will be seen later on.

²*i.e.*, the nodes $j \in V$ such that $(j, i) \in A$.

OR function.

More precisely, the reason for studying the specific case of circuits is that they are known to play an important part in the dynamics of a network. One way to see this is to note that a network whose underlying interaction graph is a tree or more generally a graph without circuits can only eventually end up in a configuration that will never change over time. A network with retroactive loops, on the contrary, will exhibit more diverse dynamical behaviour patterns. Thomas [18] had already noted the importance of underlying circuits in networks. He formulated conjectures concerning the role of positive (*i.e.*, with an even number of inhibitions) and negative (*i.e.*, with an odd number of inhibitions) circuits in the dynamics of regulation networks. Thus, from the point of view of theoretical biology as well as that of theoretical computer science (since the problem is very close to the 16th Hilbert problem concerning the number of limit cycles of dynamical systems [11]), it seems to be of great interest to address the question of the number of attractors (*i.e.*, different asymptotic dynamical behaviours) of regulation networks. In the first part of this report is given an account of the work that was done during this internship concerning the dynamics of Boolean automata circuits. More precisely, after the preliminary section 2, in sections 3 and 4, we consider, respectively, positive circuits and negative circuits updated synchronously. We obtain the exact values of the total number of attractors of these circuits and of their number of attractors of period p for every positive integer p . These values we find happen to be terms of integer sequences defined by different combinatoric problems. In section 5, we exhibit the isomorphisms that exists between some of these problems and the one we started from. Next, in the last section of part I, section 6, we complete our study of the dynamics of Boolean automata circuits by examining their behaviour when they are updated according to more general update schedules.

The second part of the work I did focused on the particular instance of (threshold) Boolean automata networks that are *OR* networks. Initially, our motivation was to understand theoretically the simulatory results presented by Elena in his PhD thesis [8]. In [8], Elena proposes a classification of networks according to their dynamics in all update schedules. This seemed to us a particularly inviting starting point for our study of networks dynamics. As the questions raised by Elena's work revealed themselves to be particularly thorny for general threshold Boolean automata networks, we decided to concentrate first on *OR* networks. Besides its simplicity, one of the main reasons for the choice of the *OR* function as transition function of networks, is the fact that it allows a certain transparency of the networks structure in the sense that it translates straightforwardly every dependency between states of automata, as we will see more concretely in part II which is the part of this document dealing with *OR* networks.

Before moving onto the body of this report, I would like to point out that in order to preserve a certain fluidity, I chose to introduce definitions and some results concerning general Boolean automata networks throughout both parts I and II even though their focus is on particular instances of these networks.

Part I

Boolean automata circuits

2 Definitions, notations and preliminary results

A **circuit** of size n is a directed graph that we will denote by $\mathbb{C}_n = (V, A)$. We will consider that its set of nodes, $V = \{0, \dots, n-1\}$, is actually $\mathbb{Z}/n\mathbb{Z}$ so that, considering two nodes i and j , $i + j$ will designate the node $i + j \pmod n$. The circuit's set of arcs is then $A = \{(i, i+1) \mid i \in \mathbb{Z}/n\mathbb{Z}\}$. Let id be the identity function ($\forall a \in \{0, 1\}, id(a) = a$) and neg the negation function ($\forall a \in \{0, 1\}, neg(a) = \neg a = 1 - a$). A *Boolean automata network* of size n associated to a circuit or **Boolean automata circuit** of size n is a couple $R_n = (\mathbb{C}_n, F)$ where $F : \{0, 1\}^n \rightarrow \{0, 1\}^n$ is the network's **global transition function**. By a minor abuse of language, we will refer to the **(global) state** of R_n as a vector $x = (x_0 \dots x_{n-1}) \in \{0, 1\}^n$ whose coefficient x_i is the state of node i of \mathbb{C}_n . The global transition function F is defined by a set of n **local transition functions** $\{f_i \in \{id, neg\} \mid i \in \mathbb{Z}/n\mathbb{Z}\}$ that will be, until section 6, applied **synchronously**: let $x = (x_0 \dots x_{n-1}) \in \{0, 1\}^n$ represent a global state of R_n , then

$$F(x) = (f_0(x_{n-1}), \dots, f_i(x_{i-1}), \dots, f_{n-1}(x_{n-2})).$$

When there will be no ambiguity as to what network we are considering, we will also note this transition rule $x(t+1) = F(x(t))$ where $x(t) \in \{0, 1\}^n$ and $t \in \mathbb{N}$ so that $\forall p \in \mathbb{N}, x(t+p) = F(F^{p-1}(x(t)))$ and at the local level of nodes, $x_i(t+1) = F(x(t))_i = f_i(x_{i-1}(t))$. Note that with the restriction on the local transition functions, $f_i \in \{id, neg\}$, we do not lose any generality. Indeed, if at least one of the nodes of the circuit, say node i , has a constant local function then its incoming arc is useless: the state of node i does not depend on that of node $i-1$ and in that case we no longer are looking at a “real” circuit. An arc $(i, i+1)$ is said to be **positive** (*resp.* **negative**) if $f_{i+1} = id$ (*resp.* $f_{i+1} = neg$). The network R_n and the circuit associated, $\mathbb{C}_n = (V, A)$, are said to be *positive* (*resp.* *negative*) if the number of negative arcs of A is even (*resp.* odd). An intuitive and very coarse explanation of such a distinction between circuits is that given a pair of consecutive inhibitions (*i.e.*, of local transition functions equal to neg) belonging to a circuit, the first member of this pair is a “real” inhibition whereas the second is only an “inhibition of the first inhibition”. Thus, positive circuits, unlike negative ones, only eventually have “neutralised” inhibitions.

Note that Boolean automata circuits as they have just been defined, are particular instances of quasi-minimal threshold Boolean automata networks³ Indeed, as it can easily be checked, id and neg can both be expressed as threshold functions and obviously, choosing local transition functions in $\{id, neg\}$ we have guaranteed that

³Quasi-minimal networks are such that if an arc is removed from their interaction graph, then their dynamics is changed.

every arc appearing in the circuit is necessary, *i.e.*, represents an existing dependency between the states of two nodes. Now, although quasi-minimal threshold Boolean automata networks were actually the starting point of the present work, we need not burden the description of the objects we will be studying with the formalism of those networks. A few additional definitions are yet needed before we can move on to the analysis of the dynamics of Boolean automata circuits.

Let $R_n = (\mathbb{C}_n, F)$ be a Boolean automata circuit of size n . In the sequel, we will make substantial use of the following function:

$$F[j, i] = \begin{cases} f_j \circ f_{j-1} \circ \dots \circ f_i & \text{if } i \leq j \\ f_j \circ f_{j-1} \circ \dots \circ f_0 \circ f_{n-1} \circ \dots \circ f_i & \text{if } j < i \end{cases}$$

There are several things to note about this function. First, because $\forall k, f_k \in \{id, neg\}$, $F[j, i]$ is injective. Second, if \mathbb{C}_n is positive then $\forall j, F[j+1, j] = id$ and if, on the contrary, \mathbb{C}_n is negative then $\forall j, F[j+1, j] = neg$. Finally, it is also important to notice that the following is true for all $t \in \mathbb{N}$, $p \leq n$, $i \in \mathbb{Z}/n\mathbb{Z}$:

$$\begin{aligned} x_i(t+p) &= f_i(x_{i-1}(t+p-1)) = f_i(f_{i-1}(x_{i-2}(t+p-2))) = \dots \\ &\dots = F[i, i-p+1](x_{i-p}(t)). \end{aligned}$$

Since the set of global states of any finite sized network R_n (may it be a Boolean automata circuit or any other type of Boolean automata network) is finite, when the network is updated, it necessarily ends up looping. In other words, $\forall x(0) \in \{0, 1\}^n$, $\exists t, p, x(t+p) = x(t)$. An **attractor** or **limit cycle** is the orbit (that is the set $\{x(t+k) \mid k \in \mathbb{N}\}$) of such a state $x(t)$. The **period** of this attractor is its cardinal, *i.e.*, the smallest p such that for any integer $k \in \mathbb{N}$, $x(t+k+p) = x(t+k)$. Elements belonging to an attractor of period 1 are usually called **fixed points**. The set of all global states of R_n belonging to an attractor of period p is denoted by:

$$\mathcal{S}_p(R_n) = \{x \in \{0, 1\}^n \mid F^p(x) = x \text{ and } \forall d < p, F^d(x) \neq x\}$$

or \mathcal{S}_p when there is no ambiguity as to what network is being considered. The number of attractors of period p of a network R_n is denoted by:

$$A_p(R_n) = \frac{1}{p} \cdot |\mathcal{S}_p(R_n)|.$$

Now, let $R_n = (\mathbb{C}_n, F)$. We define the following property \mathcal{P}_F on $\{1, \dots, n\} \times \{0, 1\}^n$ depending on F :

$$\begin{aligned} \forall p \in \mathbb{N}, \forall x \in \{0, 1\}^n, \\ \mathcal{P}_F(p, x) &\Leftrightarrow \forall i \in \mathbb{Z}/n\mathbb{Z}, x_i = F[i, i-p+1](x_{i-p}) \\ &\Leftrightarrow \forall i \in \mathbb{Z}/n\mathbb{Z}, \text{ such that } r = i \bmod p, \\ &x_r = F[r, r-p+1](x_{r-p}) \text{ and} \\ &x_i = F[i, r+1](x_r) \end{aligned}$$

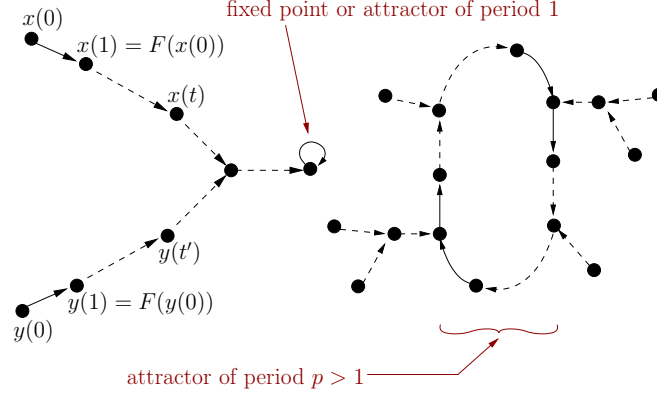


Figure 1: The **iteration graph** of an arbitrary Boolean automata network of size n . The nodes of this graph are the elements of $\{0, 1\}^n$ and an arc (x, y) exists in this graph if and only if $F(x) = y$ where F is the networks global transition function. Every strongly connected component of the iteration graph corresponds to an attractor of size the number of elements in this component.

The second equivalence above can be shown by induction on k where $i = k \cdot p + r$. The definition of $\mathcal{P}_F(p, x)$ takes its meaning from the following result which characterises global states that loop after p transitions (or less), *i.e.* states of \mathcal{S}_d where $d \leq p$:

Lemma 2.1 *Let $R_n = (\mathbb{C}_n, F)$ be a Boolean automata circuit of size n , let $p \leq n$ and let $x(0) \in \{0, 1\}^n$ be an arbitrary global state of R_n . Then,*

$$\forall t, x(t) = x(t + p) \Leftrightarrow \forall t, \mathcal{P}_F(p, x(t)).$$

Proof of Lemma 2.1 *Suppose that $\forall t, \mathcal{P}_F(p, x(t))$. Then, $\forall r, 0 \leq r < p$,*

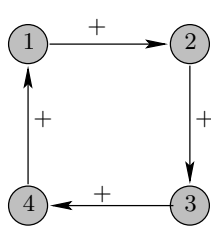
$$F[p + r, r + 1](x_r(t)) = x_{p+r}(t + p) = F[p + r, r + 1](x_r(t + p)),$$

where the first equality is always true (see the remark made above after the definition of $F[j, i]$) and the second is due to $\mathcal{P}_F(p, x(t + p))$. With the injectivity of $F[p + r, r + 1]$, this implies that $x_r(t) = x_r(t + p)$. In addition, $\forall i, p \leq i < n$ such that $r = i \bmod p$:

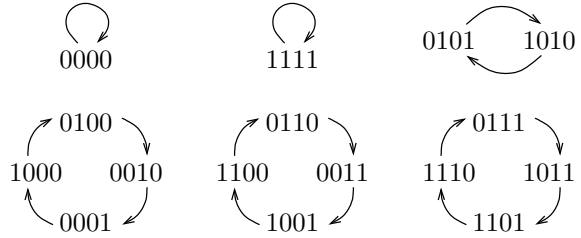
$$x_i(t + p) = F[i, r + 1](x_r(t + p)) = F[i, r + 1](x_r(t)) = x_i(t).$$

The first equality above is due to $\mathcal{P}_F(p, x(t + p))$, the second to $x_r(t + p) = x_r(t)$ and the third to $\mathcal{P}_F(p, x(t))$. On the other hand, suppose that $\forall t, x(t) = x(t + p)$. Then, $\forall i \in \mathbb{Z}/n\mathbb{Z}, x_i(t) = x_i(t + p) = F[i, i - p + 1](x_{i-p}(t))$. \square

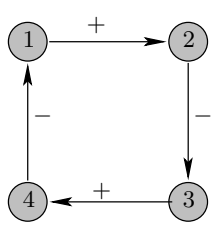
In the sequel, we will compare the dynamics of particular couples of circuits of same signs, $R_p = (\mathbb{C}_p, H)$ and $R_n = (\mathbb{C}_n, F)$, where p divides $n = p \cdot q$ and where the global transition function H of R_p is defined by the set of local transition functions



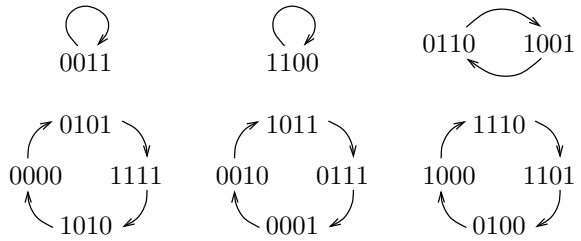
1.a.



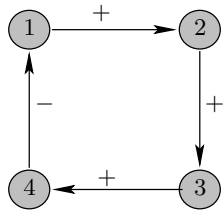
2.a.



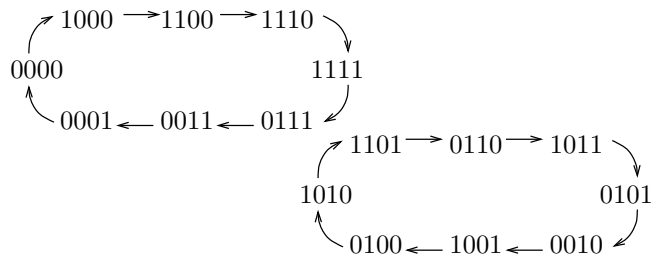
1.b.



2.b.



1.c.



2.c.

Figure 2: Figures 1.a., b. and c. represent three different Boolean automata circuits of size $n = 4$. That of figures 1.a. and b. are positive while that of figure 1.c. is negative. Figures 2.a., b. and c. picture the iteration graph of these network. Note that in all three cases here, all elements belong to an attractor. This is usually not the case with arbitrary Boolean automata networks which are not circuits.

$\{h_i \mid i \in \mathbb{Z}/p\mathbb{Z}\}$ and the global transition function F of R_n is defined by the set $\{f_i \mid i \in \mathbb{Z}/n\mathbb{Z}\}$. More precisely, we will build an isomorphism between the iteration graphs of R_n and R_p . To do this, we will use the bijection $\mathcal{Q}_{F,H}$ defined below that maps a state $x(t) \in \{0,1\}^p$ of R_p to a state $y(t) \in \{0,1\}^n$ of R_n such that $\forall c, x_0(t+c) = y_0(t+c)$. The idea behind the definition of $\mathcal{Q}_{F,H}$ is roughly to make R_n “mimic” the dynamical behaviour of R_p .

$\forall x \in \{0,1\}^p, \forall i = kp + r \in \mathbb{Z}/n\mathbb{Z}$ such that $r = i \bmod p$ and $s = i + 1 \bmod p$,

$$\begin{aligned} \mathcal{Q}_{F,H}(x)_i &= \begin{cases} x_r & \text{if } f_{i+1} = h_s \text{ and } y_{i+1} = x_s \\ & \text{or } f_{i+1} \neq h_s \text{ and } y_{i+1} \neq x_s \\ \neg x_r & \text{if } f_{i+1} = h_s \text{ and } y_{i+1} \neq x_s \\ & \text{or } f_{i+1} \neq h_s \text{ and } y_{i+1} = x_s \end{cases} \\ &= \begin{cases} x_r & \text{if } F[0, i+1] = H[0, 1]^{q-1-k} \circ H[0, s] \\ \neg x_r & \text{otherwise} \end{cases} \end{aligned}$$

The second equality can be shown by a reversed induction on i . Note that $H[0, 1] = id$ if and only R_p is positive. In the sequel, we will only consider one of the two following cases⁴:

1. R_p and R_n are both positive or
2. R_p and R_n are both negative and q is odd.

In both cases, it holds that $F[0, 1] = H[0, 1]$ and $H[0, 1]^{q-1} = id$ so that $\mathcal{Q}_{F,H}(x)_0 = x_0$. The reader can also check that $F(\mathcal{Q}_{F,H}(x))_{i+1} = H(\mathcal{Q}_{F,H}(x))_s \Leftrightarrow \mathcal{Q}_{F,H}(x)_{i+1} = x_s$ so that $F(\mathcal{Q}_{F,H}(x)) = \mathcal{Q}_{F,H}(H(x))$. Since $\mathcal{Q}_{F,H}$ is clearly bijective, in both cases mentioned, $\mathcal{Q}_{F,H}$ satisfies the desired properties.

3 Positive circuits

In this section, we focus on positive Boolean automata circuits, that is, networks associated to circuits having an even number of negative arcs. The first lemma below, which will be crucial later on, is an extension of a result proven by Goles *al.* in [9].

Lemma 3.1 *Let $R_n = (\mathbb{C}_n, F)$ be a positive Boolean automata circuit of size n . Every global state $x \in \{0,1\}^n$ of R_n belongs to an attractor of period a divisor of n .*

Proof of Lemma 3.1 *Suppose that F is defined by $\{f_i \mid i \in \mathbb{Z}/n\mathbb{Z}\}$. Let $x(t) = (x_0(t), \dots, x_{n-1}(t))$ be an arbitrary global state of R_n . Then,*

$$\forall j \in \mathbb{Z}/n\mathbb{Z}, x_j(t+n) = F[j, j+1](x_j(t+n-n)) = x_j(t).$$

The last equality above holds because, \mathbb{C}_n being positive, $F[j, j+1] = id$. □

⁴the reason why will be given further on

The purpose of the two following lemmas is to compare the dynamics of different positive circuits. More precisely, lemma 3.2 establishes an isomorphism between the attractors whose period is the largest of two networks of same size. Lemma 3.3 establishes an isomorphism between attractors of circuits of different sizes.

Lemma 3.2 *If $R_p = (\mathbb{C}_p, F)$ and $R'_p = (\mathbb{C}'_p, H)$ are two positive Boolean automata circuits, both of size p , then the number of attractors of period p of both networks is the same:*

$$A_p(R_p) = A_p(R'_p).$$

Proof of Lemma 3.2 *Let $x(t) \in \mathcal{S}_p(R_p)$ and let $y(t) = \mathcal{Q}_{F,H}(x(t))$. From the remarks concerning $\mathcal{Q}_{F,H}$ done after its definition, at the end of the previous section, $y(t+p) = \mathcal{Q}_{F,H}(x(t+p)) = \mathcal{Q}_{F,H}(x(t)) = y(t)$ and there exists no $d < p$ such that $y(t+d) = y(t)$ (otherwise, from the injectivity of $\mathcal{Q}_{F,H}$ it would hold that $x(t+d) = x(t)$ which contradicts $x(t) \in \mathcal{S}_p(R_p)$). \square*

By lemma 3.2, provided $R_p = (\mathbb{C}_p, F)$ is positive, $A_p(R_p)$ is independent of the distribution and number of negative arcs in \mathbb{C}_p . Therefore, from now on, we will use the following notation for all positive Boolean automata circuits R_p of size p :

$$A_p^+ = \frac{1}{p} \cdot |\mathcal{S}_p(R_p)| = A_p(R_p)$$

Lemma 3.3 *Let R_n be a positive Boolean automata circuit of size n . Then, for every divisor p of n ,*

$$A_p(R_n) = A_p^+.$$

Proof of Lemma 3.3 *Suppose that $R_n = (\mathbb{C}_n, F)$ where F is defined by $\{f_i \mid i \in \mathbb{Z}/n\mathbb{Z}\}$ and $n = q \cdot p$, $q, p \in \mathbb{N}$. We will show that there exists a network R_p of size p such that the sets $\mathcal{S}_p(R_n)$ and $\mathcal{S}_p(R_p)$ are isomorphic. First, we define the network $R_p = (\mathbb{C}_p, H)$ where H , the global transition function of R_p , is defined by $\{h_0 = F[0, p]\} \cup \{h_i = f_i \mid 0 < i < p\}$.*

Now, suppose that $x(t)$ belongs to $\mathcal{S}_p(R_p)$ and let $y(t) = \mathcal{Q}_{F,H}(x(t)) \in \{0, 1\}^n$. By an argument similar to that used in the proof of lemma 3.2, we find that $y(t) \in \mathcal{S}_p(R_n)$.

On the other hand, if $y(t) \in \mathcal{S}_p(R_n)$, we define $x(t) \in \{0, 1\}^p$ such that $\forall i \in \mathbb{Z}/p\mathbb{Z}$, $x_i(t) = y_i(t)$. Because \mathbb{C}_n and \mathbb{C}_p are positive and because $h_0 = F[0, p]$, $\mathcal{P}_F(p, y(t))$ (true by lemma 2.1) implies $y(t) = \mathcal{Q}_{F,H}(x(t))$. Lemma 3.1 suffices to state that $x(t+p) = x(t)$. But it is, again, by the injectivity of $\mathcal{Q}_{F,H}$ that we can claim that there is no $d < p$ such that $x(t+d) = x(t)$. \square

As a consequence of the previous lemmas, we finally get the main result of this section:

Theorem 3.1 $\forall n \in \mathbb{N}$,

$$(i) \quad 2^n = \sum_{p|n} A_p^+ \times p$$

$$(ii) A_n^+ = \frac{1}{n} \cdot \sum_{p|n} \mu\left(\frac{n}{p}\right) \cdot 2^p$$

$$(iii) T_n^+ = \frac{1}{n} \cdot \sum_{p|n} \psi\left(\frac{n}{p}\right) \cdot 2^p$$

where μ is the Möbius (see [1]) function, ψ the Euler totient function and T_n^+ the total number of distinct attractors of a positive Boolean automata circuit.

Proof of Theorem 3.1 Let R_n be a positive Boolean automata circuit. By lemma 3.1, all of the 2^n global states of a R_n belong to an attractor whose period is a divisor of n . (i) then comes from lemmas 3.2 and 3.3. (ii) is shown using the Möbius inversion formula (see [1]) on (i). For the proof of (iii), we use the fact that $\psi(m) = \sum_{r|m} (m/r) \cdot \mu(r)$:

$$\begin{aligned} T_n^+ &= \sum_{p|n} A_p^+ = \sum_{p|n} \sum_{d|p} \frac{1}{p} \cdot \mu\left(\frac{p}{d}\right) \cdot 2^d \\ &= \frac{1}{n} \cdot \sum_{p|n} \sum_{d|p} 2^d \cdot \frac{n}{p} \cdot \mu\left(\frac{p}{d}\right) = \frac{1}{n} \cdot \sum_{p|n} \sum_{d|p} 2^d \cdot \frac{n}{(p/d) \cdot d} \cdot \mu\left(\frac{p}{d}\right) \\ &= \frac{1}{n} \cdot \sum_{d|n} 2^d \sum_{k|n/d} \frac{n}{k \cdot d} \cdot \mu(k) = \frac{1}{n} \cdot \sum_{d|n} \psi\left(\frac{n}{d}\right) \cdot 2^d. \end{aligned}$$

□

In particular, point (ii) of theorem 3.1 implies that if n is prime then, since $\mu(n) = -1$,

$$A_n^+ = \frac{1}{n} \cdot (\mu(n) \cdot 2 + \mu(1) \cdot 2^n) = \frac{2^n - 2}{n}.$$

Notice also that because 1 is a divisor of all $n \in \mathbb{N}$, every positive Boolean automata circuit R_n has exactly two fixed points, *i.e.*, $A_1(R_n) = A_1(R_1) = A_1^+ = 2$. In [3] and [2], positive circuits are characterised this way and indeed, as we will see in the next section, negative circuits never have any fixed points. From this characterisation, the authors of these articles also infer some results concerning more general networks.

We performed computer simulations of the dynamical behaviour of positive circuits of sizes between 1 and 22. Simulations done for different circuits of the same size confirmed lemma 3.2. An example picturing these results is given in figure 2 where two different positive circuits of size 4 and their dynamics are represented. Table 1 shows some of the results we obtained for circuits of different sizes. In this table, n is the size of the network and p the period of the attractor. In the cell corresponding to line p and column n figures $A_p(R_n)$. Notice that as lemma 3.3 predicts, all numbers appearing on one line are the same. In particular, line one indicates that all positive circuits have two fixed points.

4 Negative circuits

We are now going to consider negative circuits. The approach we take here is very similar to that of the previous section so we will give few comments on how this

$\begin{matrix} n \\ p \end{matrix}$	1	2	3	4	5	6	7	8	9	10	11	12	21	22
1	2	2	2	2	2	2	2	2	2	2	2	2	2	2
2	-	1	-	1	-	1	-	1	-	1	-	1	-	1
3	-	-	2	-	-	2	-	-	2	-	-	2	2	-
4	-	-	-	3	-	-	-	3	-	-	-	3	-	-
5	-	-	-	-	6	-	-	-	-	6	-	-	-	-
6	-	-	-	-	-	9	-	-	-	-	-	9	-	-
7	-	-	-	-	-	-	18	-	-	-	-	-	18	-
8	-	-	-	-	-	-	-	30	-	-	-	-	-	-
9	-	-	-	-	-	-	-	-	56	-	-	-	-	-
10	-	-	-	-	-	-	-	-	-	99	-	-	-	-
11	-	-	-	-	-	-	-	-	-	-	186	-	-	186
12	-	-	-	-	-	-	-	-	-	-	-	335	-	-
21	-	-	-	-	-	-	-	-	-	-	-	-	99858	-
22	-	-	-	-	-	-	-	-	-	-	-	-	-	190557
τ_n^+	2	3	4	6	8	14	20	36	60	108	188	352	99880	190746

Table 1: Number of attractors of positive Boolean automata circuits.

case is handled. Just as lemma 3.1 does for the positive case, lemma 4.1 recalls and extends some important general properties of the dynamics of negative circuits that were mentioned by Goles *al.* in [9].

Lemma 4.1 *Let $R_n = (\mathbb{C}_n, F)$ be a negative Boolean automata circuit of size n . Then,*

1. *Every global state $x \in \{0, 1\}^n$ of R_n belongs to an attractor of period a divisor of $2n$;*
2. *If $\mathcal{S}_p(R_n) \neq \emptyset$, then p is an even divisor of $2n$ and $n = q \times \frac{p}{2}$ where q is odd.*

Proof of Lemma 4.1

1. *By a similar proof to that of lemma 3.1, we find that $\forall x(t) \in \{0, 1\}^n$, $x(n+t) = \neg x(t)$ which implies that $x(2n+t) = x(t)$. Thus every global state belongs to an attractor of period a divisor of $2n$.*
2. *Suppose that $x \in \mathcal{S}_p(R_n)$ where p divides n . By lemma 2.1, $\mathcal{P}_F(p, x)$ must be true so:*

$$\forall 0 \leq r < p, x_r = F[r, r-p+1](x_{r-p}) = F[r, r-p+1] \circ F[r-p, r+1](x_r) = F[r, r+1](x_r)$$

However, because \mathbb{C}_n is negative, $F[r, r+1] = \text{neg}$. This leads to the contradiction $x_r = \neg x_r$. Thus, if $\mathcal{S}_p(R_n) \neq \emptyset$, then p divides $2n$ without dividing n . This means that p must be even and we necessarily must have $n = q \times p/2$ where q is odd. \square

The proofs of lemmas 4.2 and 4.3 that follow are, respectively, similar to that of lemmas 3.2 and 3.3.

Lemma 4.2 *If $R_p = (\mathbb{C}_p, F)$ and $R'_p = (\mathbb{C}'_p, H)$ are two negative Boolean automata circuits, both of size p , then*

$$\forall p, A_{2p}(R_p) = A_{2p}(R'_p)$$

By lemma 4.2, provided $R_p = (\mathbb{C}_p, F)$ is negative, $A_{2p}(R_p)$ is independent of the distribution and number of negative arcs in \mathbb{C}_p . Therefore, from now on, we will use the following notation for all negative Boolean automata circuit R_p :

$$A_{2p}^- = \frac{1}{2^p} \cdot |\mathcal{S}_{2p}(R_p)| = A_{2p}(R_p).$$

Lemma 4.3 *Let R_n be a negative Boolean automata circuit of size n . Then, for every divisor p of $n = p \times q$ where q is odd,*

$$A_{2p}(R_n) = A_{2p}^-.$$

Proof of Lemma 4.3 *Suppose that $R_n = (\mathbb{C}_n, F)$ and $R_p = (\mathbb{C}_p, H)$, where H is defined as in the proof of lemma 3.3, are both negative. Suppose that $n = p \times q$ where q is odd. Then, the proof of lemma 4.3 requires to notice that if $y \in \{0, 1\}^n$ and $x = (y_0, \dots, y_{p-1}) \in \{0, 1\}^p$, it still holds that $\mathcal{P}(2p, y) \Rightarrow y = \mathcal{Q}_{F, H}(x)$. A distinction must be made between nodes $i = kp + r$ ($r = i \bmod p$) such that k is odd and nodes $i = kp + r$ such that k is even. \square*

As a consequence of the previous lemmas, we obtain the following theorem which is proven with very similar arguments to that used in the proof of theorem 3.1.

Theorem 4.1 $\forall n \in \mathbb{N}$,

$$(i) \quad 2^n = \sum_{\text{odd } q|n} A_{2n/q}^- \times 2n/q$$

$$(ii) \quad A_{2n}^- = \frac{1}{2^n} \cdot \sum_{\text{odd } q|n} \mu(q) \cdot 2^{n/q}$$

$$(iii) \quad T_n^- = \frac{1}{2^n} \cdot \sum_{\text{odd } p|n} \psi\left(\frac{n}{p}\right) \cdot 2^p$$

where T_n^- is the total number of distinct attractors of a negative Boolean automata circuit⁵.

Computer simulations of the dynamics of negative circuits of sizes between 1 and 22 were performed. The results of these simulations are shown in table 2 below (see the last paragraph of section 3 for an explanation of what holds each cell of this table). A particular case of formulas (ii) and (iii) of theorem 4.1 is when $n = 2^k$. Then, since 1 is the only odd divisor of n , $A_{2n}^- = T_n^- = 2^{n-k-1}$ (see cells ($p = 16, n = 8$) and ($p = 32, n = 8$) of table 2).

Again, we may also note that theorem 4.1 implies that a negative circuit never has any fixed points in its dynamics. In [3] and [2], the authors proven a result stating that *arbitrary* networks containing only negative circuits have no fixed points.

⁵and where μ and ψ still are, respectively, the Möbius function and the Euler totient

$p \backslash n$	1	2	3	4	5	6	7	8	15	16	17	18	21	22
2	1	-	1	-	1	-	1	-	1	-	1	-	1	-
4	-	1	-	-	-	1	-	-	-	-	-	1	-	1
6	-	-	1	-	-	-	-	-	1	-	-	-	1	-
8	-	-	-	2	-	-	-	-	-	-	-	-	-	-
10	-	-	-	-	3	-	-	-	3	-	-	-	-	-
12	-	-	-	-	-	5	-	-	-	-	-	5	-	-
14	-	-	-	-	-	-	9	-	-	-	-	-	9	-
16	-	-	-	-	-	-	-	16	-	-	-	-	-	-
30	-	-	-	-	-	-	-	-	1091	-	-	-	-	-
32	-	-	-	-	-	-	-	-	-	2048	-	-	-	-
34	-	-	-	-	-	-	-	-	-	-	3855	-	-	-
36	-	-	-	-	-	-	-	-	-	-	-	7280	-	-
42	-	-	-	-	-	-	-	-	-	-	-	-	49929	-
44	-	-	-	-	-	-	-	-	-	-	-	-	-	95325
T_n^-	1	1	2	2	4	6	10	16	1096	2048	3856	7286	49940	95326

Table 2: Number of attractors of negative Boolean automata circuits.

5 Related problems

Sequences $(A_n^+)_{n \in \mathbb{N}}$, $(T_n^+)_{n \in \mathbb{N}}$, $(A_{2n}^-)_{n \in \mathbb{N}}$ and $(T_n^-)_{n \in \mathbb{N}}$ defined in the previous sections happen to correspond precisely to the integer sequences A1037, A31, A48 and A16, respectively, of the OEIS [16]. In [16], these sequences are defined by different combinatoric problems. In this section, we will study how some of them are related to the problem of counting the number of attractors (of given period or in total) of Boolean automata circuits. Conveniently, the relationships that we find between these different but equivalent problems, provide different ways of formalising the dynamics of Boolean automata networks updated synchronously.

First of all, note that thanks to the results of the previous sections we may concentrate our attention on one *canonical* positive (*resp.* negative) circuit, wisely chosen. Indeed, since A_p^+ (*resp.* A_p^-) does not depend on the distribution or number of negative arcs in the circuit in question as long as this number is even (*resp.* odd), we may choose a circuit $R_n = (C_n, F)$ as the representative of all positive (*resp.* negative) circuits of size n . Then, for any other positive (*resp.* negative) network $R'_n = (C'_n, F')$ of size n , there exists a permutation σ of $\{0, 1\}^n$ such that $\forall x \in \{0, 1\}^n, F'(x) = F(\sigma(x))$. In the positive case, the choice of a *canonical* circuit is straightforward. We will choose the positive circuit with no negative arcs and denote it by $R_n^+ = (C_n, F^+)$. Note that F^+ is defined by a set of n local functions all equal to *id* and acts as a rotation of the coefficients of vectors in $\{0, 1\}^n$:

$$\forall (x_0, \dots, x_{n-1}) \in \{0, 1\}^n, F^+(x_0, \dots, x_{n-1}) = (x_{n-1}, x_0, \dots, x_{n-2}).$$

The *canonical* negative circuit is less obvious to choose because a circuit of size n with n negative arcs is negative only if n is odd. Thus, unless mentioned otherwise, we will be calling canonical negative network of size n , the network $R_n^- = (C_n, F^-)$

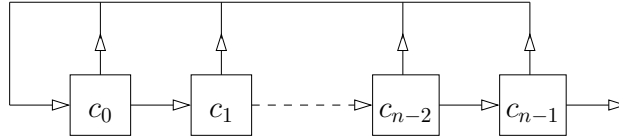
whose circuit has 1 negative arc (the arc $(n-1, 0)$) and $n-1$ positive arcs. In that case, it holds that:

$$\forall (x_0, \dots, x_{n-1}) \in \{0, 1\}^n, F^-(x_0, \dots, x_{n-1}) = (\neg x_{n-1}, x_0, \dots, x_{n-2}).$$

5.1 Binary shift register sequences

Let f be a function of $\{0, 1\}^n \rightarrow \{0, 1\}$. A **n -stage binary shift register** is a group of n cells, c_0, \dots, c_{n-1} , connected together. If, at time t , cell c_i contains the binary variable x_i , $i \in \mathbb{Z}/n\mathbb{Z}$, then at time $t+1$, x_{n-1} is outputted, the variables x_0, \dots, x_{n-2} are shifted to the right so that cell c_i contains x_{i-1} and $f(x_0 \dots x_{n-1})$ is inserted in cell c_0 . An output sequence of a shift register is the series of its outputs, *i.e.*, a sequence of the form:

$$\dots x_{n-1}, x_{n-2}, \dots, x_0, f(x_0 \dots x_{n-1}), f(f(x_0 \dots x_{n-1}), x_1 \dots x_{n-1}) \dots$$



Following Golomb [10] and Sloane [15], we will consider two types of shift registers. The first one corresponds to $f(x_0 \dots x_{n-1}) = x_0$ and the second to $f(x_0 \dots x_{n-1}) = \neg x_0$. The relationship between shift registers and Boolean automata circuits is then fairly obvious. If the binary variable that cell c_i contains at time t is seen as the state $x_i(t)$ of node i in the circuit \mathbb{C}_n , then, a n -stage shift register of the first type corresponds to R_n^+ and the n -stage shift register of the second type corresponds to R_n^- . In both cases, f corresponds to f_0 and the shift of the $n-1$ other cells corresponds to the $f_i = id$, $0 < i < n$.

Sloane denotes by $Z(n)$ the number of different possible output sequences from a shift register of the first type and by $Z^*(n)$ that of the second type. Thus, $Z(n)$ also denotes the total number of attractors (of any period) of a positive Boolean automata circuit, \mathbb{T}_n^+ , and $Z^*(n)$ denotes the same number for negative Boolean automata circuit, \mathbb{T}_n^- . Theorem 3.1 of [15] states that

$$Z(n) = A31(n) = \frac{1}{n} \cdot \sum_{d|n} \psi(d) \cdot 2^{n/d}$$

$$\text{and } Z^*(n) = A16(n) = \frac{1}{2n} \cdot \sum_{\text{odd } d|n} \psi(d) \cdot 2^{n/d}$$

which do, indeed, unsurprisingly, match our formulas for \mathbb{T}_n^+ and \mathbb{T}_n^- respectively. The interesting point lies in the proof of the first expression. Following Golomb, we

may see global states of R_n^+ as elements of a set on which acts the cyclic group of order n generated by the permutation $\pi = (1, \dots, n)$ (which actually is precisely the rotation F^+). This way, the orbits $\{\pi^i(x) \mid x \in \{0, 1\}^n, i \in \mathbb{Z}/n\mathbb{Z}\}$ correspond to the attractors of the network. And since π^i can be written as the product of $\gcd(n, i)$ cycles of length $\text{ord}(\pi^i) = n/\gcd(n, i)$, the number of vectors x such that $\pi^i(x) = x$, *i.e.* the number of configurations belonging to an attractor of period a divisor of i , is $2^{\gcd(n, i)}$. The Burnside Lemma [6] then allows to determine the number, $Z(n)$, of orbits/attractors. The second expression is derived from the first.

5.2 Binary necklaces and Lyndon words

A binary n -bead necklace [14] is defined by a word x of size n on the alphabet $\{0, 1\}$ or any rotation of this word. Let us call $[x]$ the set of all words which are rotations of one another and which define the same n -bead necklace. The sets $[x]$ are obviously equivalence classes. The primitive period of a necklace defined by the words in $[x]$ is the smallest number of times one needs to turn the necklace around before returning to a configuration that is undistinguishable from its initial configuration. In other words, the primitive period of a necklace defined by the words in $[x]$ is $|[x]|$. It also is the smallest $p \in \mathbb{N}$ such that any $x' \in [x]$ can be written $x' = y^p$, where $y \in \{0, 1\}^{n/p}$ and $y^p = yy^{p-1}$. The number of different necklaces of size n , *i.e.*, of different sets $[x]$, is known [16] to be equal to $T_n^+ = A31(n) = Z(n)$. Again, since F^+ acts as a rotation, it is easy to see that the sets $[x]$ correspond to attractors of R_n^+ and their primitive periods correspond to the periods of those attractors.

A **binary Lyndon word** of length n is the lexicographically least word of a class $[x]$ whose primitive period is n . There are $A_n^+ = A1037(n)$ such words and thus, in every attractor of period n of R_n^+ , there is exactly one global state x corresponding to a Lyndon word of size n (all others are non-trivial rotations of x).

The negative case is trickier. First, in [16], only the sequence $A48(n) = A_{2n}^-$ has an entry related to necklaces ($A16(n) = T_n^-$ has not). According to this entry, $A48(n) = A_{2n}^-$ equals the number of unlabelled binary Lyndon words. A **n -bead unlabelled necklace** corresponds to a set $[x] \cup [-x]$. An **unlabelled binary Lyndon word** of size n is the lexicographically least word of a set $[x] \cup [-x]$ where $|[x]| = |[-x]| = n$. When n is odd, it is easy to see the relationship between these words and the attractors of a negative network $R_n = (\mathbb{C}_n, F)$ whose circuit has n negative arcs and none positive. For such a circuit, the following equalities hold:

$$\forall x = (x_0, \dots, x_{n-1}) \in \{0, 1\}^n, F(x) = (-x_{n-1}, -x_0, \dots, -x_{n-2}) = \neg F^+(x),$$

$$F^k(x) = \begin{cases} F^{+k}(x) & \text{if } k \text{ is even} \\ \neg F^{+k}(x) & \text{if } k \text{ is odd} \end{cases}$$

$$\text{and in particular, } \forall k \in \{0, \dots, n\}, F^{k+n}(x) = F^k(F^n(x)) = F^k(-x),$$

where F^+ is still the global transition function of the canonical positive circuit R_n^+ mentioned above. On the other hand, because n is odd, there exists no class $[x]$ such that $\neg x \in [x]$ so $[x] \cap [-x] = \emptyset$. Thus, sets corresponding to unlabelled necklaces are

of the form:

$$[x] \cup [-x] = \{F^{+k}(x) \mid 0 \leq k < n\} \cup \{F^{+k}(-x) \mid 0 \leq k < n\} = \{F^k(x) \mid 0 \leq k < 2n\}.$$

which is exactly the definition of the attractors of R_n . In each of the sets $[x] \cup [-x]$ where the primitive period of x is n , by definition, there is exactly one unlabelled binary Lyndon word of primitive period n . Consequently, there are as many such words as there are attractors of period $2n$ in the dynamics of the negative Boolean automata circuit with no positive arcs R_n .

When n is even, however, there generally exists sets $[x]$ and $[-x]$ that are equal so that the sets $[x] \cup [-x]$ do not coincide with the attractors of the negative Boolean automata circuit R_n .

5.3 Cycles in a digraph under $x^2 \bmod q$

In this section, we are going to present yet another point of view on the dynamics of positive circuits inspired by the work presented by Shallit and Vasiga in [19]. In this article, the authors study, amongst other things, the cycles in a digraph under $x^2 \bmod q$. In the framework of the cyclic multiplicative group $(\mathbb{Z}/q\mathbb{Z})^*$, where $q = 2^{n+1} - 1$ is a Mersenne prime, they find an expression for the total number of cycles, of elements in the cycles and of elements that are not in any cycle. Here, we carry out a very similar study in the cyclic additive group $G = \mathbb{Z}/(2^n - 1)\mathbb{Z}$. This change of group allows, in particular, to avoid having any elements of even order which happen to be precisely those who, in $(\mathbb{Z}/q\mathbb{Z})^*$, do not belong to any cycle (see [19]).

If we see the global states of the canonical positive Boolean automata circuit network R_n^+ as numbers between 0 and $2^n - 1$ written in binary ($x = \overline{x_0 \dots x_{n-1}}^2 = \sum_{i=0}^{n-1} x_i \cdot 2^i$), then the global transition function of R_n^+ can be written

$$F^+(x) = (2 \cdot x \bmod 2^n) + x_{n-1}$$

and it holds that for any $c \in \mathbb{N}$: $F^{+c}(x) = (2^c \cdot x \bmod 2^n) \cdot (1 - 2^{-n}) + x \cdot 2^{c-n}$. Thus, for any $c \in \mathbb{N}$, $c \leq n$ the following holds (where the ‘‘sufficient direction’’ of the last but least equivalence can be checked easily using the fact that $c \leq n$):

$$\begin{aligned} F^{+c}(x) = x &\Leftrightarrow 0 = (2^c \cdot x \bmod 2^n) \cdot (1 - 2^{-n}) + x \cdot (2^{c-n} - 1) \\ &\Leftrightarrow 0 = (2^c \cdot x \bmod 2^n) \cdot (2^n - 1) + x \cdot (2^c - 2^n) \\ &\Leftrightarrow x \cdot (2^c - 2^n) = x \cdot (2^c - 1 + 1 - 2^n) \equiv 0 \bmod 2^n - 1 \\ &\Leftrightarrow x \cdot (2^c - 1) \equiv 0 \bmod 2^n - 1 \quad (1) \end{aligned}$$

Now, let x be an element of the cyclic group $G = \mathbb{Z}/(2^n - 1)\mathbb{Z}$ and let $d = \text{ord}(x) = \min\{k \mid x \cdot k \equiv 0 \bmod 2^n - 1\}$ be its order in this group. Suppose that $x \in \mathcal{S}_p(R_n^+)$. Then, $p = \min\{0 < c \mid F^{+c}(x) = x\} = \min\{0 < c \leq n \mid x \cdot (2^c - 1) \equiv 0 \bmod 2^n - 1\}$ (the first equality is by definition of $\mathcal{S}_p(R_n^+)$ and the second comes from (1) above). Therefore, either $p = 1$, and $x \equiv 0 \bmod 2^n - 1$, *i.e.*, $x \in \{0, 2^n - 1\}$. Either $p > 1$ and

then, by definition of d , it holds that $x \cdot (2^p - 1) \equiv 0 \pmod{2^n - 1} \Leftrightarrow 2^p \equiv 1 \pmod{d}$. In that case, if $\text{ord}_d(2)$ is the order of 2 in the group $(\mathbb{Z}/d\mathbb{Z})^*$, then $p = \text{ord}_d(2)$ and

$$\mathcal{S}_p(R_n^+) = \{x \in G \mid \text{ord}(x) = d \text{ and } p = \text{ord}_d(2)\}, \quad p > 1.$$

Every one of the $\psi(d)$ elements in G of order d is thus in a cycle of period $\text{ord}_d(2)^6$. Using the fact that when $p = \text{ord}_d(2)$, $d \mid 2^n - 1$ if and only if $p \mid n$, we can now count the total number of cycles or attractors:

$$\sum_{p \mid n} \frac{1}{p} \cdot |\mathcal{S}_p(R_n^+)| = \sum_{d \mid 2^n - 1} \frac{\psi(d)}{\text{ord}_d(2)}$$

Noticing that necessarily every element $x \in G$ belongs to a cycle and using the Möbius inversion formula again, we can show that this last expression is equal to the integer sequences already met $A31(n)$ and T_n^n . We also may note that the number of each type of cycle depends very little on the circuit itself : it depends on n , the size of the circuit, in that the sizes of the cycles must divide n but except from that it otherwise only depends on the Euler's totient function.

6 Synchronous, sequential and block sequential update schedules

We now will look at general update schedules that do not necessarily update all nodes of a network synchronously. In this section we will continue to focus on Boolean automata circuits. However, the definition of update schedule is general to all Boolean networks.

An **update schedule** or *u.s.* is a function $s : V \rightarrow \{0, \dots, n-1\}$, where V is the set of nodes of the network and n its size. $s(i)$ represents the *date* of update of node i within one global time step at the end of which all nodes of the network and thus the network have been updated. Without loss of generality, we will consider only *u.s.s* s such that $\min\{s(i) \mid i \in V\} = 0$ and $\forall t, 0 \leq t < n-1, \exists i \in V, s(i) = t+1 \Rightarrow \exists j \in V, s(j) = t$. The synchronous *u.s.* considered in the previous sections is denoted by π . It is such that $\forall i \in V, \pi(i) = 0$. A **sequential** *u.s.* s is such that $\forall i, j \in V, s(i) \neq s(j)$, *i.e.*, $\forall t, 0 \leq t < n, \exists i \in V, s(i) = t$. There are $n!$ different sequential *u.s.s* of a set of n nodes. *U.s.s* that are neither synchronous nor sequential are called **block-sequential**. For an arbitrary *u.s.* s , we will write $\mathcal{B}_t^s = \{i \in V \mid s(i) = t\}$. Since s defines an ordered partition of V , we may use the notation suggested in the example 6.1. By [7], we know that the number of different *u.s.s* of a set of n nodes is given by the following formula :

$$\mathcal{I}_n = \sum_{k=0}^{n-1} \binom{n}{k} \mathcal{I}_k.$$

⁶ ψ still is Euler's totient function

The number of different *u.s.s* of a networks nodes, thus, grows exponentially with the size of the network.

Example 6.1 Let $V = \{0, \dots, 5\}$. The function $s : V \rightarrow \{0, \dots, 5\}$ such that $s(5) = 0, s(3) = 1, s(1) = 2, s(0) = 3, s(2) = 4$ and $s(4) = 5$ is a sequential *u.s.*. The function $r : V \rightarrow \{0, \dots, 5\}$ such that $r(2) = 0, r(3) = r(4) = 1$ and $r(0) = r(1) = r(5) = 2$ is a block sequential *u.s.*. A more practical way of denoting s, r and the synchronous *u.s.* is the following:

$$s \equiv (5)(3)(1)(0)(2)(4) \quad r \equiv (2)(3, 4)(0, 1, 5) \quad \pi \equiv (0, 1, 2, 3, 4, 5).$$

Let $R_n = (\mathbb{C}_n, F)$ be an arbitrary Boolean automata circuit of size n and s an *u.s.* of its nodes. F is defined by the set of local transition functions $\{f_i \mid i \in \mathbb{Z}/n\mathbb{Z}\}$. We define the global transition function with respect to s denoted by F^s in the following manner:

$$F^s : \begin{cases} \{0, 1\}^n & \rightarrow \{0, 1\}^n \\ x(t) & \mapsto F^s(x(t)) = (f_1^s(x(t)), \dots, f_{n-1}^s(x(t))) \end{cases}$$

where

$$f_{i+1}^s(x(t)) = \begin{cases} f_{i+1}(x_i(t)) & \text{if } s(i) \geq s(i+1) \\ f_{i+1}(x_i(t+1)) & \text{if } s(i) < s(i+1) \end{cases}$$

In the sequel, we will write $x^s(t+1)$ instead of $F^s(x(t))$ when there will be no ambiguity as to what global transition function F we are considering. We define the *inversions* of s with respect to \mathbb{C}_n as the set

$$inv(s) = \{(i, i+1) \mid s(i) < s(i+1)\}.$$

For nodes of an inversion $(i, i+1)$, $x_{i+1}(t+1)$ depends on $x_i(t+1)$ instead of $x_i(t)$ as is the case when $s(i+1) \leq s(i)$ and when, in particular, $s = \pi$. Obviously, the number of inversions of an *u.s.* of a circuit of size n is strictly smaller than n . The only *u.s.* that has no inversions is the synchronous *u.s.* π . In the sequel, given a Boolean automata circuit R_n , we will refer to the dynamics induced by the *u.s.* s as the dynamics of R_n when its nodes are updated according s . Now, let

$$i^* = \max\{k < i \mid (k, k+1) \notin inv(s)\}, \quad \forall i \in \mathbb{Z}/n\mathbb{Z},$$

$$A^s = \{(i, j) \mid \text{such that } x_j^s(t+1) \text{ depends on } x_i^s(t)\} = \{(i^*, i) \mid i \in \mathbb{Z}/n\mathbb{Z}\},$$

$\mathbb{C}_n^s = (\mathbb{Z}/n\mathbb{Z}, A^s)$ and finally let R_n^s be the network (\mathbb{C}_n^s, F^s) where the global transition function F^s is defined above but is also given, in the usual manner, by the set of local transition functions $\{h_i = F[i, i^* + 1] \mid i \in \mathbb{Z}/n\mathbb{Z}\}$. Then, when the nodes of R_n^s are updated synchronously, the dynamics of this network are identical to that of R_n updated according to s . Furthermore, if $|inv(s)| = k$, then the nodes $i \in \mathbb{Z}/n\mathbb{Z}$ such that $\exists j \in \mathbb{Z}/n\mathbb{Z} \ i = j^*$ belong to a subgraph \mathbb{C}_{n-k} of \mathbb{C}_n^s which is a circuit of size $n - k$ and of sign that of \mathbb{C}_n . All nodes not belonging to \mathbb{C}_{n-k} depend on one and only one node of \mathbb{C}_{n-k} (as in figure 3).

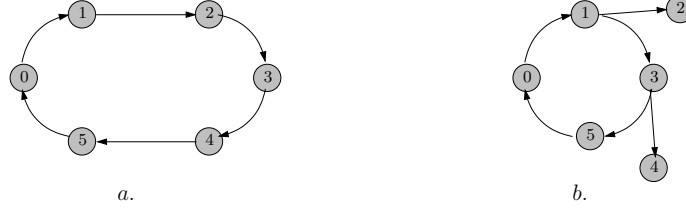


Figure 3: *a.* The underlying interaction graph \mathbb{C}_6 of a network $R_6 = (\mathbb{C}_6, F)$ where F is arbitrary. *b.* The interaction graph \mathbb{C}_6^s associated to R_6^s where $s \equiv (2)(3, 4)(0, 1, 5)$. $\text{inv}(s) = \{(2, 3), (4, 5)\}$. The underlying circuit \mathbb{C}_4 in \mathbb{C}_6^s has as set of nodes $\{0, 1, 3, 5\} = \{i \in \mathbb{Z}/6\mathbb{Z} \mid \exists j \in \mathbb{Z}/6\mathbb{Z} i = j^*\}$.

Before moving on to the results of this section, let us first recall that if the dynamics of a network has fixed points for a certain *u.s.*, then it has the same fixed points for every other *u.s.*.

Proposition 1 *Let $R_n = (\mathbb{C}_n, F)$ be a Boolean automata circuit of size n and let r and s be two of its *u.s.s.**

- (i) *The dynamics induced by r and by s are identical if and only if $\text{inv}(r) = \text{inv}(s)$.*
- (ii) *Furthermore, if $\text{inv}(r) \neq \text{inv}(s)$, then the dynamics induced by s and by r have no attractor of period $p > 1$ in common.*
- (iii) *If $|\text{inv}(s)| = k$, then the dynamics of R_n are isomorphic to that of a Boolean automata circuit R_{n-k} of size $n - k$.*

Proof of Proposition 1

(i) *follows directly from theorem... of [4].*

(ii) *Let us suppose that $\text{inv}(r) \neq \text{inv}(s)$ and the dynamics induced by s and by r share an attractor of period $p > 1$, $\mathcal{A} = \{x(t), \dots, x(t+p-1)\}$: $\forall a, x(t+a) = x^s(t+a) = x^r(t+a)$. Let $(i, i+1) \in \text{inv}(s) \setminus \text{inv}(r)$. By definition of i^* , it holds that $\forall a \in \mathbb{N}$,*

$$x_i^s(t+a+1) = F[i, i^*+1](x_{i^*}(t+a))$$

$$\text{and } x_{i+1}^s(t+a+1) = f_{i+1}(x_i^s(t+a+1)) = F[i+1, i^*+1](x_{i^*}(t+a)).$$

Thus, we can write the following:

$$\begin{aligned} x_{i+1}(t+2) &= x_{i+1}^s(t+2) = F[i+1, i^*+1](x_{i^*}(t+1)) = \\ &= x_{i+1}^r(t+2) = f_{i+1}(x_i(t+1)) = f_{i+1}(x_i^s(t+1)) = \\ &= f_{i+1} \circ F[i, i^*+1](x_{i^*}(t)) = F[i+1, i^*+1](x_{i^*}(t)). \end{aligned}$$

By the injectivity of $F[i + 1, i^* + 1]$, this implies that $x_{i^*}(t + 1) = x_{i^*}(t)$ and thus that the state of node i^* and, consequently, that of all nodes are constant. A must then be a fixed point. This contradicts $p > 1$.

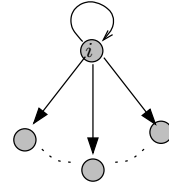
(iii) follows immediately from the structure of \mathbb{C}_n^s described above. □

Following proposition 1, we may define the equivalence relation between *u.s.s* that relates r and s if and only if $\text{inv}(s) = \text{inv}(r)$. We will denote by $[s]$ the equivalence class of s . The number of distinct equivalence classes of *u.s.s* with k inversions, is clearly $\binom{n}{k}$. As a direct consequence, the following result holds:

Proposition 2 *Let $R_n = (\mathbb{C}_n, F)$ be a Boolean automata circuit of size n . The total number of distinct dynamics induced by the *u.s.s* of R_n is*

$$\sum_{k=0}^{n-1} \binom{n}{k} = 2^n - 1.$$

Let $R_n = (\mathbb{C}_n, F)$ be a Boolean automata circuit of size n . We may note that a particular type of equivalence class is that of classes containing *u.s.s* with $n - 1$ inversions. There are n such classes. Each one is characterised by the unique $i \in \mathbb{Z}/n\mathbb{Z}$ that is such that $(i, i + 1)$ is *not* an inversion. Each class contains exactly one *u.s.* which is sequential, namely, the *u.s.* $s^i \equiv (i+1)(i+2) \dots (i-1)(i)$ where $\text{inv}(s^i) = \{(j, j+1), j \neq i\}$. The figure on the right pictures the graph underlying the network $R_n^{s^i}$. Because there is a loop over node i in this graph, the dynamics induced by s^i contains only fixed points if \mathbb{C}_n is positive and only attractors of period 2 if \mathbb{C}_n is negative.



Proposition 3 *In all $[s]$, $s \neq \pi$, there exists a sequential *u.s.*.*

Proof of Proposition 3 *Consider a Boolean automata circuit of size n . By definition of the equivalence classes, it is enough to prove that for every set of $k < n$ inversions, there exists a sequential *u.s.* that satisfies exactly these k inversions. Thus, let $\text{inv}(r)$ such that $|\text{inv}(r)| = k$. We define the sets consecutive integers $I_l = \{i^l, i^l + 1, \dots, i^l + |I_l| - 1\}$ and $K_l = \{i^l + |I_l|, i^l + |I_l| + 1, \dots, i^{l+1} - 1\}$, where we suppose that the indices l belong to a group $\mathbb{Z}/m\mathbb{Z}$ so that in particular $m + 1 = 0$. The sets I_l and K_l are such that $\mathbb{Z}/n\mathbb{Z} = \bigcup_{l \leq m} (I_l \cup K_l)$ and:*

$$\forall i \in I_l, (i, i + 1) \in \text{inv}(r), \quad \forall i \in K_l, (i, i + 1) \notin \text{inv}(r).$$

Using the definitions of these sets, we now define a sequential *u.s.* s that has exactly the k inversions of $\text{inv}(r)$ as the reader can check :

1. $s(i^m) = 0$

2. $\forall l \in \mathbb{Z}/m\mathbb{Z}, s(i^{l-1}) = s(i^l) + |I_l|$
3. $\forall l \in \mathbb{Z}/m\mathbb{Z}, \forall k < |I_l|, s(i^l + k + 1) = s(i^l + k) + 1$
4. $s(i^m + |I_m|) = n - 1$
5. $\forall l \in \mathbb{Z}/m\mathbb{Z}, s(i^{l-1} + |I_{l-1}|) = s(i^l + |I_l|) + |K_l|$
6. $\forall l \in \mathbb{Z}/m\mathbb{Z}, \forall k < |K_l|, s(i^l + |I_l| + k + 1) = s(i^l + |I_l| + k) + 1$

Notice that points 3 and 6 above guarantee, respectively, that $\forall i \in I_l, (i, i+1) \in \text{inv}(s)$ and $\forall i \in K_l, (i, i+1) \notin \text{inv}(s)$ so that $\text{inv}(s) = \text{inv}(r)$. \square

Example 6.2 Let $s \equiv (2)(3,4)(0,1,5)$ be an u.s. of the nodes of \mathbb{C}_6 (as in figure 3....3). Its set of inversions is $\text{inv}(s) = \{(2,3), (4,5)\}$. According to proposition 1 (i), all of the following u.s.s induce the same dynamics as s , i.e., belong to $[s]$:

$$s_1 \equiv (4,2)(0,1,3)(5), \quad s_2 \equiv (4)(2)(3)(1)(0)(5), \quad s_3 \equiv (2)(4)(1)(0)(5)(3),$$

$$s_4 \equiv (4)(2)(1)(0)(3)(5).$$

s_2 corresponds to the sequential u.s. built in the proof of proposition 3 when $I_1 = \{2\}$, $I_2 = \{4\}$, $K_1 = \{3\}$, and $K_2 = \{1,0,5\}$. s_3 corresponds to the sequential u.s. built in the proof of proposition 3 when $I_1 = \{4\}$ and $I_2 = \{2\}$. s_4 is not constructed as in the proof of proposition 3.

Proposition 4 Given a Boolean automata circuit (positive or negative) $R_n = (\mathbb{C}_n, F)$ and a set $\mathcal{A} = \{x(0), \dots, x(p-1)\}$, $p > 1$, we can determine in $\mathcal{O}(p \cdot n)$ steps whether there exists an u.s. s such that \mathcal{A} is an attractor of period p of R_n^s , i.e., $F^s(x(t)) = x(t+1), \forall 0 \leq t < p$ and $F^s(x(p-1)) = x(0)$. If such an u.s. s exists, we can compute in $\mathcal{O}(p \cdot n)$ steps a sequential u.s. $r \in [s]$.

Proof of Algorithm 1 To prove that algorithm 1 does indeed return the required u.s. when it exists, we must prove that the sets I_l and K_l that it computes are the same as those defined with the same names in the proof of proposition 3. Suppose s is the u.s. inducing the attractor \mathcal{A} . First note that $(i-1, i) \notin A(s) \Leftrightarrow (i-1, i) \in \text{inv}(s)$ Now, $\exists t, x_i(t) \neq y_i(t) \Rightarrow f_i \neq f_i^s = F[i, i^*+1] \Leftrightarrow i^* \neq i-1 \Rightarrow (i-1, i) \notin A(s)$. On the other hand, suppose that $(i-1, i) \notin A(s)$ that is, $i^* \neq i-1$. There exists necessarily (because $p > 1$) an integer t such that $x_{i^*}(t) \neq x_{i^*}(t+1)$ so that $x_i(t+2) = F[i, i^*+1](x_{i^*}(t+1)) \neq y_i(t+2) = F(x(t+1))_i = f_i(x_{i-1}(t+1)) = f_i \circ F[i-1, i^*+1](x_{i^*}(t))$. \square

In conclusion of this first part of my report, I may say that most combinatoric problems concerning Boolean automata circuits have now been dealt with. In particular, we know the exact value of both the total number of attractors and the number of attractors of period p , $\forall p \in \mathbb{N}$, in the dynamics of positive and negative Boolean automata circuits updated with the synchronous, sequential and the block sequential

Algorithm 1: Finding a sequential *u.s.* that induces a particular attractor of a given Boolean automata circuit if it exists

Input: $R_n = (\mathbb{C}_n, F)$ and $\mathcal{A} = \{x(0), \dots, x(p-1)\}$ as in proposition 4.

Output: A sequential *u.s.* s such that the attractor \mathcal{A} belongs to the dynamics of R_n^s if such an *u.s.* exists.

```

begin
1   In  $\mathcal{O}(p \cdot n)$  steps, compute the set
     $\mathcal{A}^\pi = \{y(t) \mid 0 \leq t \leq p, y(t+1) = F(x(t))\}$  (where  $y(0) = y(p)$ );
2   In  $\mathcal{O}(p \cdot n)$  steps, compute the sets of consecutive integers  $I_l$ 
    and  $K_l$  as well as their sizes  $|I_l|$  and  $|K_l|$  such that  $\forall i \in I_l,$ 
     $\exists t < p, x_i(t) \neq y_i(t)$  and  $\forall i \in K_l, \forall t < p, x_i(t) = y_i(t), \forall i \in K_l, \forall j \in I_l,$ 
     $i > j$ ;
3   In  $\mathcal{O}(n)$  steps, compute the sequential u.s.  $s$  as in the proof
    of proposition 3 using the sets  $I_l$  and  $K_l$ ;
4   In  $\mathcal{O}(p \cdot n)$  steps, compute the set
     $\mathcal{A}^s = \{x(0), F^s(x(0)), \dots, F^{s^{p-1}}(x(0))\}$  and check that  $\mathcal{A}^s = \mathcal{A}$ . If
    not then no u.s. induces  $\mathcal{A}$  as an attractor.
end

```

update schedules. We also know how many different dynamics can be induced by the set of update schedules of a Boolean automata circuit. One main question remains unanswered however : “What are the sizes of the equivalence classes of update schedules that yield the same dynamics?”. For the very particular cases of $[\pi]$ and of the classes of update schedules with $n - 1$ inversions (where n is the size of the circuit) we know the size of the classes is 1. We also obtained a very intricate formula for the size of classes of update schedules having *consecutive* inversions only. Being particularly hard to analyse, I chose not to let this formula feature in this report. It however implies that the sizes of such classes is exponential as may certainly be that of many other classes. One motive (amongst others) for studying this question follows from A. Elena’s work. In his PhD thesis [8], Elena computed statistics of the number of attractors of threshold Boolean automata networks as well as of their periods averaging over all networks (of sizes between 3 and 6) and all update schedules. For both he found particularly small values. Now, as I have already mentioned, it is known that underlying circuits play an important role in the dynamics of a network with an arbitrary structure. Provided we knew how to link some way the dynamics of a circuit with that of an arbitrary network containing it, knowing the answer to this question would thus allow us to understand whether the averages found by Elena are small because most networks contain only small circuits or if it is because most update schedules of the networks restricted to their underlying circuits correspond to those equivalence classes inducing the “same” dynamics as that of very small circuits updated synchronously.

Equivalence classes of u.s.s consecutive inversions [part removed in final report submitted]

Lemma 6.1 *Let $V = U \cup W$ where $U = \bigcup_{t < q} \mathcal{B}_t$ and $W = \{w_1, \dots, w_k\}$. The number of u.s.s s of V satisfying :*

- (i) $\forall i \in \mathcal{B}_{t_i}, \forall j \in \mathcal{B}_{t_j}, s(i) = s(j) \Leftrightarrow t_i = t_j$ and $s(i) < s(j) \Leftrightarrow t_i < t_j$,
- (ii) $\forall i \leq k, s(w_i) < s(w_{i+1})$

is equal to $\mathcal{Z}(q, k) = \mathcal{Z}(q, k-1) + 2 \cdot \sum_{t=0}^q \mathcal{Z}(q-t, k-1)$.

Proof of *By induction on k and q . If $k = 0$ then, the only u.s. s satisfying the desired conditions is such that $s(i) = t, \forall i \in \mathcal{B}_t$. If $q = 0$, then, the only u.s. s satisfying the desired conditions is such that $s(w_i) = i$. Suppose that $k, q \geq 1$. There are three cases. In the first one, there exists $t < q$ such that $\forall t' \leq t, \forall i \in \mathcal{B}_{t'} s(i) = t'$ and $s(w_1) = t$. Then, $\forall i > 1, s(w_i) > t$ so that there are $\mathcal{Z}(q-t, k-1)$ such u.s.s. In the second case, there exists $t < q$ such that $\forall t' \leq t, \forall i \in \mathcal{B}_{t'} s(i) = t'$ and $s(w_1) = t+1$. Again, there are $\mathcal{Z}(q-t, k-1)$ such u.s.s. In the third case, $s(w_1) = 0$ and $\forall i \in \mathcal{B}_0 s(i) > 0$. There are $\mathcal{Z}(q, k-1)$ Summing over t , the result follows. \square*

Proposition 5 *Let $[s]$ be an equivalence class of u.s.s that have k consecutive inversions, i.e., $\forall r \in [s], \exists i, j < n,$*

$$\forall k < i, \forall k \geq j, (k, k+1) \notin \text{inv}(r) \quad \text{and} \quad \forall k, i \leq k < j, (k, k+1) \in \text{inv}(r).$$

$$\text{Then, } |[s]| = \sum_{q=1}^{n-k} \binom{n-k}{q} \times \mathcal{Z}(q, k-1).$$

Proof of $\binom{n-k}{q}$ *is the number of ways of partitioning the set $\{i < n \mid (i, i+1) \notin \text{inv}(s)\}$ into q parts. The reason why $\mathcal{Z}(q, k-1)$ figures in this formula instead of $\mathcal{Z}(q, k)$ is that the image of node $i = \min\{k \mid (k, k+1) \in \text{inv}(s)\}$ is necessarily 0. \square*

Part II

OR networks

In this section we are going to look at networks that are more general in their structure than the circuits considered above, but less general in their transition function since all of their local transition functions are equal to the function OR with no negated entries. We will call this class of networks OR^+ . A network R of OR^+ is thus completely defined by its underlying interaction graph $G = (V, A)$. This is why, in the sequel, with a slight abuse of language we will refer to G indifferently as a *graph* or as a *network* of OR^+ . Let i be a node of such a network $G = (V, A)$ of size $|V| = n$ and let $\mathcal{N}^-(i) = \{j \in V \mid (j, i) \in A\}$. Then, if $x(t) = (x_1(t), \dots, x_n(t))$ is the global state of G at time t , with the synchronous *u.s.*, the local state of i at time $t + 1$ is:

$$x_i(t + 1) = \bigvee_{j \in \mathcal{N}^-(i)} x_j(t),$$

i.e., $x_i(t + 1) = 1 \Leftrightarrow \exists j \in \mathcal{N}^-(i), x_j(t) = 1$. It is interesting to note that this can be expressed in terms of matrices. Let M be adjacency matrix of G , that is, the $n \times n$ matrix such that $\forall i, j, M_{ij} = 1 \Leftrightarrow (i, j) \in A$. Then $\forall x(t) \in \{0, 1\}^n, \forall k \in \mathbb{N}, x(t+k) = x(t) \cdot M^k$. Without loss of generality, we suppose all networks considered in this section to be connected. Many notations used in the sequel are obvious extensions of notations introduced in the previous sections.

In his PhD Thesis [8], Adrien Elena defines four classes of Boolean threshold automata networks and describes the extensive computer simulations he performed in particular to give statistics of the number of networks in each class. The four classes are defined below. First, note again in this framework that the set of fixed points of the dynamics of a network is invariant with *u.s.* changes.

1. **Fi** is the class of networks whose dynamics only has fixed points whatever the *u.s.*,
2. **Cy** is the class of networks whose dynamics only has attractors of period $p > 1$ whatever the *u.s.*,
3. **Mi** is the class of networks whose dynamics has at least one fixed point and one attractor of period $p > 1$ whatever the *u.s.*,
4. **Ev** is the class of networks whose dynamics has fixed points for all *u.s.s* and for some *u.s.s*, not all, it also has attractors of period $p > 1$.

For networks of OR^+ of arbitrary size n , it is clear that the states⁷ 0^n and 1^n are always fixed points. Consequently, there are no OR^+ networks in the class Cy. In order to carry out a theoretical study related to Elena's work in the particular case of OR^+ networks we define in table 3 a variant of the set of four classes above. Clearly, every OR^+ network belongs to one of the six classes defined in this table. In the

⁷If, $a \in \{0, 1\}$, $x = a^n$ is the vector such that $\forall i, x_i = a$.

	$\exists FP 0^n \text{ et } 1^n$	$\exists \text{ d'autres } FP \neq 0^n, 1^n$	$\exists CL_p, p > 1$
Fi	O	N	N
Fi'	O	O	N
Cy	O	N	O
Mi	O	O	O
Ev	O	N	$\exists m.i. \text{ tel que } O$ $\exists m.i. \text{ tel que } N$
Ev'	O	O	$\exists m.i. \text{ tel que } O$ $\exists m.i. \text{ tel que } N$

Table 3: In the first column of this table, figures the name of the classes that are defined on the corresponding lines. In the first line figures properties of the dynamics of networks: a “Y” in the second column means that the networks dynamics have the two fixed points 0^n and 1^n , a “Y” in the second column means that the networks dynamics have other fixed points, a “N” means they have not and finally, in the last column, a “Y” means that the networks dynamics have attractors of period $p > 1$ and a “N” means they have not. “ $\exists u.s.$ such that Y” and “ $\exists u.s.$ such that N” that appear in the last columns of lines corresponding to the classes Ev and Ev’ mean that for some $u.s.s$ networks in these classes have attractors of period $p > 1$ and for some they have not. From theorem 6.3, the lines in white correspond to classes that are empty. From corollary 6.1, the lines in red correspond to classes of OR^+ networks that are strongly connected and lines in blue to networks that are not strongly connected.

sequel, given two nodes of a network u and v , we call *walk* of size k from u to v a set of nodes $\{v_0 = u, v_2, \dots, v_k = v\}$ where $\forall i < k, (v_i, v_{i+1})$ is an arc of the network. The following lemma is easy to show by induction and is crucial to the other results of this section:

Lemma 6.2 *Let G be an OR^+ network and $i, j \in G$ such that there exists a walk of size k from i to j in G . Then, $x_i(t) = 1 \Rightarrow x_j(t + k) = 1$.*

This lemma has two important consequences. The first one is that if a node v with a loop (*i.e.*, an arc from v to v) is initially in state 1 then it will always stay in that state. The second is that if there exists a node in a strongly connected component C whose state is fixed at 1 (for instance because there is a loop on it or simply because we are considering a fixed point in which 1 is this nodes state) then the state of this component will necessarily evolve to (or remain in) the state $1^{|C|}$. Hence the following results:

Proposition 6 *Let G be an OR^+ network of arbitrary size n . Let C be a strongly connected component of G and let $x \in \{0, 1\}^n$ be a fixed point of G . Then, either $\forall i \in C, x_i = 0$ or $\forall i \in C, x_i = 1$.*

Corollary 6.1 *An OR^+ network of arbitrary size n has as sole fixed points 0^n and 1^n , i.e., belongs to one of the classes Fi , Cy and Ev , if and only if it is strongly connected. As a consequence, networks belonging to the classes Fi' , Mi and Ev' have at least two strongly connected components.*

Proof of Corollary 6.1 *The first direction of the equivalence comes from proposition 6. Suppose $G = (V, A)$ is an OR^+ network of size n with at least two strongly connected components. Let $G^* = (\mathcal{C}, \mathcal{A})$ be the graph whose set of nodes \mathcal{C} is the set of strongly connected components of G and whose set of arcs is $\mathcal{A} = \{(B, C) \mid B, C \in \mathcal{C}, \exists i \in B, \exists j \in C, (i, j) \in A\}$. Let $B, C \in \mathcal{C}$ such that $(B, C) \in \mathcal{A}$ and let $\mathcal{U} = \{D \in \mathcal{C} \mid \exists \text{ a walk in } G^* \text{ from } D \text{ to } B\}$ and $\mathcal{D} = \{D \in \mathcal{C} \mid \exists \text{ a walk in } G^* \text{ from } C \text{ to } D\}$. By definition of G^* , these two sets are disjoint. Then, the state $x \in \{0, 1\}^n$ of G satisfying $\forall i \in \mathcal{U}, x_i = 0$ and $\forall i \in \mathcal{D}, x_i = 1$ is clearly a fixed point of G different from 0^n and from 1^n . \square*

We are now going to look at different update schedules of OR^+ networks. In a network $G = (V, A)$ updated according to an update schedule s , the following holds:

$$x_i(t+1) = 1 \Leftrightarrow \exists j \in \mathcal{N}^-(i), [s(j) \geq s(i) \wedge x_j(t) = 1] \vee [s(j) < s(i) \wedge x_j(t+1) = 1]$$

In the sequel, we will denote by $G(s)$ the network (or graph) corresponding to the network (or graph) G updated according to an arbitrary u.s. s , that is, the graph in which the arc (i, j) exists if and only if $x_j^s(t+1)$ depends on $x_i^s(t)$. In particular, if $s = \pi$, then $G(\pi) = G$. We will refer to the dynamics of G updated according to s as that of $G(s)$ (updated according to π). The following lemma which is easy to prove, is true independantly of the type of transition function of the network :

Lemma 6.3 *Let $G = (V, A)$ be the graph associated to a Boolean network. Then, $(u, v) \in G(s)$ if and only if there exists a walk $\{v_0 = u, \dots, v_k = v\}$ from u to v in G such that*

$$s(u) \geq s(v_1) \text{ and } \forall i, 1 \leq i < k, s(v_i) < s(v_{i+1}).$$

As a consequence, we have the following result :

Theorem 6.1 *For any OR^+ network G there exists an u.s. s such the dynamics of $G(s)$ has only fixed points.*

Proof of Theorem 6.1 *For every strongly connected component C of G we can choose a node $v \in C$ and a walk $\{v_0 = v, v_1, \dots, v_k = v\}$ from v to v . We then can define an u.s. s such that for every such node and such walk, $s(v_{i+1}) > s(v_i)$, $\forall 0 < i < k$ so that there exists a loop over node v in $G(s)$. Then, from the previous results, the dynamics of $G(s)$ has only fixed points. \square*

Now, we define the following property \mathcal{P} that applies to strongly connected graphs or components of a graph :

$$\mathcal{P}(G) \Leftrightarrow \eta(G) = 1$$

where $\eta(G)$ is the greatest common divisor of the lengths of all circuits in the strongly connected graph or component G . It is known (see for instance [5]) that this property is equivalent to the adjacency matrix of G being *primitive*. Results mentioned in [5] allow us to derive the following theorem:

Theorem 6.2 *Let G be a strongly connected graph and s an arbitrary u.s.. $\mathcal{P}(G(s))$ is true if and only if the dynamics of $G(s)$ has no attractor of period $p > 1$.*

Proof of Theorem 6.2 *As it has been mentioned above theorem 6.2, $\mathcal{P}(G(s))$ is equivalent to the adjacency matrix $M(s)$ of $G(s)$ being primitive. Now it is also known (again, see [5]) that the primitivity of $M(s)$ is equivalent to the existence of an integer N such that $\forall p \geq N$, $M(s)^p$ is a positive matrix. This means that if $x(t) \neq 0^n$ then there exists an integer p such that $x^s(t+p) = x(t) \cdot M(s)^p = 1^n$. On the contrary, if $M(s)$ is imprimitive then such an integer N does not exist. The greatest common divisor of the lengths of all circuits in $G(s) = (V, A)$ is an integer $\eta(G(s)) = k > 1$ and the set of nodes can be partitioned the following way (the proof of this is given in [5]) : $V = \bigcup_{i < k} V_i$ where $\forall (u, v) \in A$, $\exists i < k$, $u \in V_i$, $v \in V_{i+1}$ and $\forall v \in V_{i+1}$, $\exists u \in V_i$, $(u, v) \in A$. Then, any $x \in \{0, 1\}^n$, $x \neq 0^n$, $x \neq 1^n$, such that $\forall i < k$, $\forall u, v \in V_i$, $x_u = x_v$ belongs to an attractor of period $p > 1$. \square*

Summing up the previous results and their consequences, we have the following:

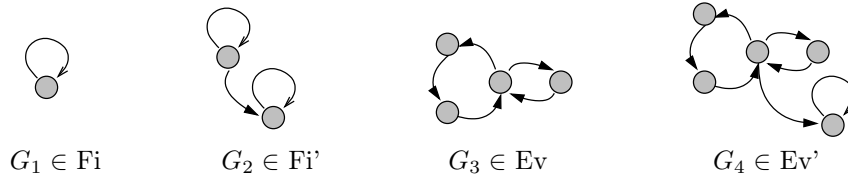
Theorem 6.3 (i) $M_i = C_y = \emptyset$,

(ii) Any strongly connected graph G such that $\neg \mathcal{P}(G)$ is true, belongs to Ev ,

(iii) Any graph G such that every one of its strongly connected components C satisfies $\neg \mathcal{P}(C)$ belongs to Ev' ,

(iv) None of the classes Fi , Fi' , Ev , Ev' is empty.

Proof of Theorem 6.3 (i) follows from theorem 6.1, (ii) and (iii) from theorems 6.1 and 6.2 and (iv) from (ii), (iii) and the fact that all networks with at least one loop in each of their strongly connected components belongs either to Fi or to Fi' according to the number of these components. \square



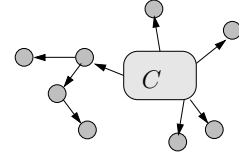
The remaining question which needs to be answered in order to be able to classify all OR^+ networks into one of the six classes of table 3 is the following :

Which are the OR^+ networks G for which there exists an u.s. s such that the dynamics of $G(s)$ has attractors of period $p > 1$?

Following theorem 6.3, the remaining unclassified networks are those that have (or are) strongly connected components C without loops such that $\mathcal{P}(C)$ is true. These networks whose dynamics only has fixed points for the synchronous *u.s.*, belong either to Fi or to Ev (*resp.* to Fi' or to Ev') if they are strongly connected (*resp.* if they are not). Before, identifying sub-classes of these networks belonging to one of the classes Fi, Fi', Ev and Ev', let us first determine some properties of the graphs $G(s)$.

Proposition 7 *Let G be a strongly connected graph and s an *u.s.* of its nodes. Then, $G(s)$ has one unique strongly connected component C . If $u \in C$, then $\forall v \notin C$, there is no walk from v to u and if $v \notin C$, then for a certain $u \in C$, there exists a walk from u to v .*

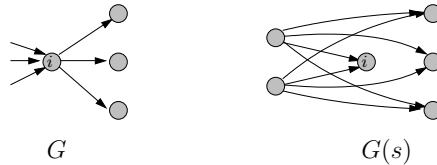
Proof of Proposition 7 *Because, in $G = (V, A)$, $\text{deg}^-(u) = |\mathcal{N}^-(u)| > 0, \forall u \in V$ the same is necessarily true in $G(s) = (V, A(s))$ (with both *u.s.* π and s , every nodes state depends on at least one other nodes state). Thus, there exists a strongly connected component in $G(s)$ and for any node a which is not in a strongly connected component, there exists a walk to a from such a component. Now, suppose there exists two distinct strongly connected components C_1 and C_2 in $G(s)$ and let $u \in C_1$ and $v \in C_2$. In G there exists a walk $\{v_0 = u, v_1, \dots, v_k = v\}$ from u to v . For every i such that there exists a $j < i$ satisfying $s(v_j) \geq s(v_{j+1})$, it can be proven that there exists a walk from v_j to v_i using lemma 6.3. If i is such that there is no such j then $\forall k < i, s(v_k) < s(v_{k+1})$. In that case, $\forall k \leq i, x_{v_k}^s(t+1)$ depends on $x_u^s(t+1)$ which depends on the $x_w(t)$ where w is such that $(w, u) \in A(s)$ so that also $(w, v_k) \in A(s)$. Among such nodes w there necessarily exists some that belong to C_1 . Therefore, in any case, there exists a walk from a node $w \in C_1$ to v_i and thus, as well, a walk from u to v_i in $G(s)$. Consequently, in $G(s)$, there exists a walk from u to v and, for the same reason, a walk from v to u . This contradicts the existence of the two distinct strongly connected components C_1 and C_2 .*



□

In the following result, $\mathcal{N}^+(i) = \{j \in V \mid (i, j) \in A\}$ and $\text{deg}^+(i) = |\mathcal{N}^+(i)|$.

Proposition 8 *Let $G = (V, A)$ be a strongly connected graph and s an *u.s.* of its nodes. Then $G(s)$ is strongly connected if and only if $\forall i \in V, \exists j \in \mathcal{N}^+(i), s(i) \geq s(j)$.*



Proof of Proposition 8 *If $i, j \in V$ are such that $j \in \mathcal{N}^+(i)$ and $s(i) \geq s(j)$, then $x_j^s(t+1)$ depends on $x_i^s(t)$, $\forall x(t)$ so that $(i, j) \in A(s)$ and $\text{deg}^+(i) > 0$ in $G(s)$.*

Thus, if $\forall i \in V, \exists j \in \mathcal{N}^+(i), s(i) \geq s(j)$, then $\forall i \in V, \text{deg}^+(i) > 0$ so that all nodes belong to the sole (cf proposition 7) strongly component of $G(s)$. Conversely, if i is such that $\forall j \in \mathcal{N}^+(i), s(j) > s(i)$ then, there are no $k \in V$ such that $x_k(t+1)$ depends on $x_i(t)$ and $\text{deg}^+(i) = 0$ in $G(s)$. \square

Now, we examine a few particular classes of OR^+ networks and determine to which of the four non-empty classes defined in table 3 they belong.

Proposition 9 *Let $G = (V, A)$ be a symmetric graph ($\forall u, v \in V, (u, v) \in A \Rightarrow (v, u) \in A$). G belongs to Fi if and only if it contains a circuit of odd length. Otherwise, $G = G(\pi)$ has attractors of period 2 and $\forall s \neq \pi, G(s)$ has only fixed points.*



Figure 4: Two symmetric graphs as in proposition 9. G_1 has a circuit of odd length so $G_1 \in Fi$. G_2 only has circuits of even length so $G_2 \in Ev$.

Proof of Proposition 9 *For any nodes u and v such that $(u, v), (v, u) \in A$, if $s(u) > s(v)$ then $x_u^s(t+1)$ depends on $x_v^s(t+1)$ which depends on $x_u^s(t)$ so that there exists a loop over u in $G(s)$. So the only u.s. of G that can induce limit cycles is the synchronous u.s. π . If there exists an odd length circuit in G , then since there also exists circuits of length two in G (between any pair of connected nodes), $\mathcal{P}(G)$ is true. \square*

In the following result, *edges* are defined as walks that can only intersect at their extremities *Circuit-edges* are edges that belong to at least one circuit of the graph considered. The length of an edge is the number of arcs in it.

Proposition 10 *A graph that has at least one strongly connected component and only has circuit-edges of length $l \geq 2$ belongs to Ev or to Ev' (according to how many strongly connected components it has).*

Proof of Proposition 10 *First note that the property $\mathcal{P}(G(s))$ does not depend on arcs that are not in a circuit. For every circuit-edge $\{v_0, v_1, \dots, v_l\}$ of length $l \geq 2$ let s be an u.s. satisfying :*

$$s(v_{i+1}) \leq s(v_i), \forall i \in \{1, \dots, l-1\} \text{ and } s(v_3) \leq s(v_2) \text{ if and only if } l \text{ is even.}$$

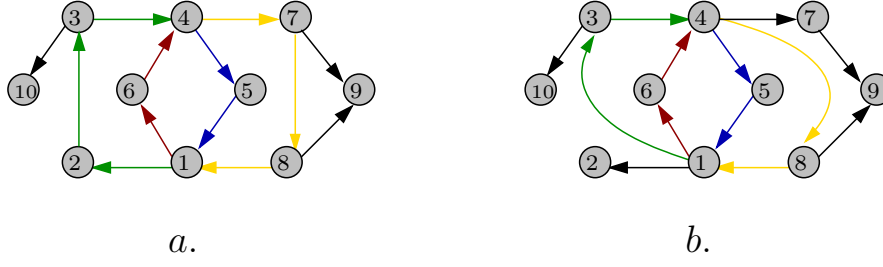


Figure 5: *a.* A graph G with as that considered in proposition 10. Different colours correspond to different circuit-edges. This graph has circuits of sizes 4, 5 and 6 so that $\eta(G) = 1$. *b.* The graph $G(s)$ where $s \equiv (1, 2, 4, 5, 6, 7, 9, 10)(3, 8)$. The circuits in $G(s)$ all have size 4 = $\eta(G(s)) > 1$.

Note that if $l = 2$ then $\forall i \leq 2, s(v_{i+1}) \leq s(v_i)$ and the walk remains unchanged in $G(s)$. Otherwise, if l is odd $s(v_3) > s(v_2)$ has as only consequences that in $G(s)$, v_2 no longer belongs to a circuit-edge and the length of the edge between v_1 and v_l is cut down to $l - 1$ without changing the length of any other circuit-edge since $\mathcal{N}^-(v_3) = \{v_2\}$. Finally, by definition of circuit-edges, such an *u.s.* exists. In $G(s)$ all circuit-edges are then of even length so that the circuits also all have an even sizes and $\neg\mathcal{P}(G(s))$. \square

In fact, the construction of the proof of proposition 10 may give an *u.s.* s such that $\eta(G(s)) > 1$ for networks G that are less constrained than those mentioned in proposition 10 (see figure below where $s \equiv (2, 3)(1, 4)$).



Let $G = (V, A)$ be a graph of size $|V| = n$. We call *covering of type k* of G a set $\mathcal{C} = \bigcup_{\mathbb{C}} \mathcal{C}^{\mathbb{C}}$ of nodes of G where the \mathbb{C} s are circuits of G , the sets $\mathcal{C}^{\mathbb{C}}$ are such that: (i) $\mathcal{C}^{\mathbb{C}} \subseteq \mathbb{C}$, (ii) $\exists q \in \mathbb{N}, |\mathcal{C}^{\mathbb{C}}| = k \cdot q$ and (iii) if $\mathbb{C} = \{v_0, \dots, v_{p-1}\}$ and $\mathbb{C}' = \{v'_0, \dots, v'_{r-1}\}$ are two distinct circuits of G then $\forall v = v_i = v'_j \in \mathbb{C} \cap \mathbb{C}', v \in \mathcal{C}^{\mathbb{C}}, v \notin \mathcal{C}^{\mathbb{C}'}, v_{i+1} \neq v'_{j+1}$. The purpose of the following result is to allow to express problems concerning the dynamics of OR^+ networks with respect to different *u.s.s* without mentioning *u.s.s*.

Proposition 11 *Let $G = (V, A)$ be a graph. There exists an *u.s.* s such that $\eta(G(s)) = k > 1$ if and only if there exists a covering of type k of G .*

To prove proposition 11, the following lemma is needed. It can easily be proven using the results of the previous section or lemma 6.3.

Lemma 6.4 Let $G = (V, A)$ be a graph and s an u.s. of its nodes.

- (i) If $\mathbb{C} = \{v_0, \dots, v_{p-1}\}$ is a circuit of G , then $\mathbb{C}^s = \{v_i \in \mathbb{C}, s(v_i) \geq s(v_{i+1})\}$ is a circuit of $G(s)$;
- (ii) If \mathbb{C}^s is a circuit of $G(s)$, then there exists a circuit \mathbb{C} of G such that all nodes of \mathbb{C}^s belong to \mathbb{C} .

Proof of Lemma 6.4 First suppose the u.s. s is such that $\eta(G(s)) = k > 1$ and for every $\mathbb{C} = \{v_0, \dots, v_{p-1}\}$ in G , let $\mathbb{C}^{\mathbb{C}} = \{v_i \in \mathbb{C}, s(v_i) \geq s(v_{i+1})\}$ and let $\mathcal{C} = \bigcup_{\mathbb{C}} \mathbb{C}^{\mathbb{C}}$. Then from lemma 6.4, the sets $\mathbb{C}^{\mathbb{C}}$ are circuits of $G(s)$ so that k is a divisor of their size. The last condition required for \mathcal{C} to be a covering of type k of G is obviously satisfied. On the other hand, suppose $\mathcal{C} = \bigcup_{\mathbb{C}} \mathbb{C}^{\mathbb{C}}$ is a covering of type k of G . Let s be an u.s. such that for any circuit $\mathbb{C} = \{v_0, \dots, v_{p-1}\}$ of G , $\forall v_i \in \mathbb{C}^{\mathbb{C}}, s(v_i) > s(v_{i+1}) \forall v_i \in \mathbb{C} \setminus \mathbb{C}^{\mathbb{C}}, s(v_i) \leq s(v_{i+1})$ (it may be verified that such a s always exists). Then, the sets $\mathbb{C}^{\mathbb{C}}$ are clearly the circuits of $G(s)$. \square

From proposition 11, follows proposition 12 in which the generalised lattice is a graph as in figure 6. Except on the border of the lattice, nodes are of in-degree and out-degree either both one or both two and all squares of the lattice are circuits. On each edge, there can be an arbitrary number of nodes greater or equal to two⁸. By

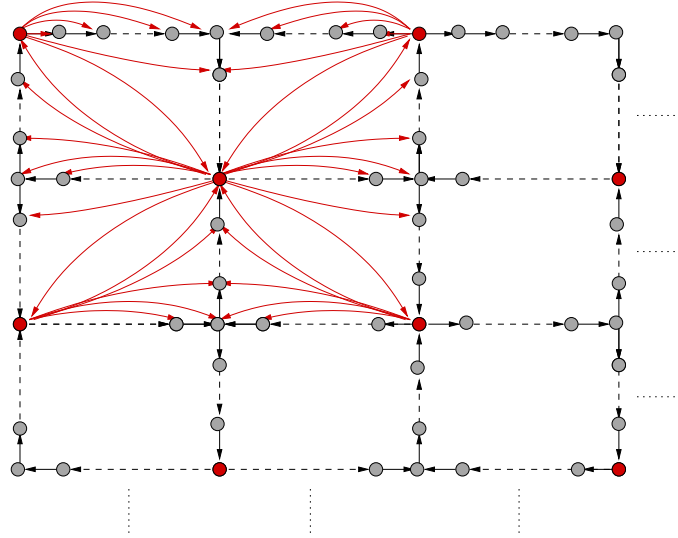


Figure 6: The generalised lattice. The arcs in black are the arcs of the initial graph G . Arcs in red belong to $G(s)$ where s is constructed as in the proof of proposition 11 with \mathcal{C} being the set of nodes in red.

showing that there exists a cover of type 2 as in figure 6, we can build an u.s. s such that $G(s)$ has attractors of period 2.

⁸Note that the generalised lattice does not fall into the scope of proposition 10 when it has edges with just two nodes, *i.e.*, one arc.

Proposition 12 *The generalised lattice belongs to Ev.*

From the study presented in this section on OR^+ networks the first thing we have learned is that however simple these networks seem to be, they give rise to many non trivial problems. Thus, it seems reasonable to think that performing the same study for arbitrary threshold Boolean automata networks still is a rather ambitious task at this stage. Indeed, a property of OR^+ networks we have made implicit and extensive use of in this section is that of having dynamics that depend very tightly on the underlying structure of the network. Generally this property is not true for other networks. Being able to justify theoretically the results Elena found by computer simulations on small networks, and in particular, being able to account for the sizes of the classes he defined as well as for the average period of their attractors, thus remains a particularly inviting challenge, very closely related to the problem examined in part I of this report. However, although our aim is effectively to understand the dynamics of threshold Boolean automata networks and not just OR^+ networks⁹, completing our understanding of the latter would probably be a first useful step. To do this, we would need to answer the following question which is the main one remaining unanswered, at least for arbitrary OR^+ networks, at the end of part II:

Amongst the networks that have no loops and are such that all of their strongly connected components C satisfy $\mathcal{P}(C)$, which are the ones that have attractors of period $p > 1$ for a certain update schedule?

7 Conclusion

To sum up very briefly, during this internship, on the first hand I studied Boolean automata circuits and significantly cut into the combinatorial analysis of their dynamics in all update schedules. The remaining relevant question in this framework concerns the number of different update schedules inducing the same dynamics, given a circuit. Subsequently, rather than choosing to go on studying particular patterns that may be found underlying in arbitrary networks, such as circuits, I chose to lift all restrictions on the structure of the networks. However, in order to manage to classify networks according to their dynamics, as was my aim, some different types of restrictions were needed. Thus, OR^+ networks with their very simple transition function were chosen and some progress that may be pursued was made in their classification.

Beyond the questions that arise directly from the two different studies I carried out during this internship (that of parts I and II respectively) and besides their obvious need for an extension to more general networks than the one examined here, I believe this work calls for many other investigations in the same or in close lines.

⁹or, incidently, AND^+ networks which can be treated almost identically by exchanging the role of states 1 and 0

A first natural and essential perspective would be to carry out some comparisons with other related studies and the results they produced. For instance, in [12], [3] and [8], experimental or theoretical results prove or suggest that the networks in question have only very little different asymptotic dynamical behaviours ($\mathcal{O}(\sqrt{n})$ in the case of connectivity 2 networks considered in [12] and [3], one or two in the case of the small networks studied by Elena). If, again, we suppose the underlying circuits in a network play a decisive part in its dynamics, this seems to be at first sight in contradiction with the exponential number of attractors of Boolean automata circuits that we found in part I. I believe it would be interesting to connect the two sources of results in order to lift the contradiction.

Another immediate angle of inquiry that could be taken would be to explore further the dynamics of networks and consider not only their attractors but also their *basins of attraction*, *i.e.*, the set of global configurations that lead to a particular attractor of a network.

Acknowledgements. Besides Jacques Demongeot and Eric Goles who supervised and guided my work during the past months, many thanks are also owed to Sylvain Sené, Ivan Rappaport, Julio Aracena and Éric Rémila who all contributed to the work presented in this document.

References

- [1] T. M. Apostol. *Introduction to analytic number theory*. Springer-Verlag, 1976.
- [2] J. Aracena. *Discrete mathematics models associated to biological systems. Application to the genetic regulatory networks*. PhD thesis, Université Joseph Fourier - Grenoble, 2001.
- [3] J. Aracena, S. Ben Lamine, O. Mermet, O. Cohen, and J. Demongeot. Mathematical modeling in genetic networks: relationships between the genetic expression and both chromosomal breakage and positive circuits. *IEEE Transactions on Systems, Man, and Cybernetics*, 33:825–834, 2003.
- [4] J. Aracena, E. Goles, and L. Salinas. On the robustness of update schedules in boolean networks. *Biosystems (submitted)*, 2008.
- [5] R. A. Brualdi and H. J. Ryser. *Combinatorial matrix theory*. Cambridge University Press, 1991.
- [6] W. Burnside. *Theory of groups of finite order*. Cambridge University Press, 1911.

- [7] J. Demongeot, A. Elena, and S. Señ. Robustness in regulatory networks: a multi-disciplinary approach. *Acta Biotheoretica*, 56(1-2):27–49, 2008.
- [8] A. Elena. *Robustesse des réseaux d’automates booléens à seuil aux modes d’itération. Application à la modélisation des réseaux de régulation génétique*. PhD thesis, Université Joseph Fourier - Grenoble, 2009.
- [9] E. Goles, F. Fogelman-Soulie, and G. Weisbuch. Specific roles of different boolean mappings in random networks. *Bulletin of Mathematical Biology*, 44(5):715–730, 1982.
- [10] S. W. Golomb. *Shift register sequences*. Aegean Park Press, Laguna Hills, CA, USA, 1981.
- [11] D. Hilbert. Mathematical problems. *Bulletin of the American Mathematical Society*, 8:437–479, 1902.
- [12] S. A. Kauffman. Metabolic stability and epigenesis in randomly constructed genetic nets. *Journal of Theoretical Biology*, 22:437–467, 1969.
- [13] W. S. McCulloch and W. Pitts. A logical calculus of the ideas immanent in nervous activity. *Journal of Mathematical Biology*, 5:115–133, 1943.
- [14] F. Ruskey. Information on necklaces, unlabelled necklaces, Lyndon words, De Bruijn sequences.
- [15] N. J. A. Sloane. On single-deletion-correcting codes. In K. T. Arasu and A. Seress, editors, *Codes and Designs*, pages 273–291. de Gruyter, Berlin, 2002.
- [16] N. J. A. Sloane. The On-Line Encyclopedia of Integer Sequences, 2008.
- [17] R. Thomas. Boolean formalisation of genetic control circuits. *Journal of Theoretical Biology*, 42:563–585, 1973.
- [18] R. Thomas. On the relation between the logical structure of systems and their ability to generate multiple steady states or sustained oscillations. *Springer Series in Synergetics*, 9:180–193, 1981.
- [19] T. Vasiga and J. Shallit. On the iteration of certain quadratic maps over $\text{GF}(p)$. *Discrete Mathematics*, 277:219–240, 2004.