HAL
open science

On the Effects of Firing Memory in the Dynamics of Conjunctive Networks

Eric Goles, Pedro Montealegre, Martín Ríos-Wilson

To cite this version:

Eric Goles, Pedro Montealegre, Martín Ríos-Wilson. On the Effects of Firing Memory in the Dynamics of Conjunctive Networks. 25th International Workshop on Cellular Automata and Discrete Complex Systems (AUTOMATA), Jun 2019, Guadalajara, Mexico. pp.1-19, 10.1007/978-3-030-20981-0_1 . hal-02312604

HAL Id: hal-02312604

https://inria.hal.science/hal-02312604v1

Submitted on 11 Oct 2019

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.

Distributed under a Creative Commons Attribution 4.0 International License

On the effects of firing memory in the dynamics of conjunctive networks.

Eric Goles[1], Pedro Montealegre[1], and Martín Ríos-Wilson[2]

[1] Facultad de Ingeniería y Ciencias, Universidad Adolfo Ibáñez, Santiago, Chile.
eric.chacc@uai.cl,p.montealegre@uai.cl
[2] Departamento de Ingeniería Matemática, Facultad de Ciencias Físicas y Matemáticas, Universidad de Chile, Santiago, Chile. mrios@dim.uchile.cl

Abstract. Boolean networks are one of the most studied discrete models in the context of the study of gene expression. In order to define the dynamics associated to a Boolean network, there are several *update schemes* that range from parallel or *synchronous* to *asynchronous.* However, studying each possible dynamics defined by different update schemes might not be efficient. In this context, considering some type of temporal delay in the dynamics of Boolean networks emerges as an alternative approach. In this paper, we focus in studying the effect of a particular type of delay called *firing memory* in the dynamics of Boolean networks. Particularly, we focus in symmetric (non-directed) conjunctive networks and we show that there exist examples that exhibit attractors of non-polynomial period. In addition, we study the prediction problem consisting in determinate if some vertex will eventually change its state, given an initial condition. We prove that this problem is **PSPACE**-complete.

Keywords: Boolean network · Firing memory · Conjunctive networks. · Prediction problem. · PSPACE.

1 Introduction

Boolean networks are one of the simplest and most studied discrete models in the context of the study of gene expression [25,24,34]. A boolean network is defined by a boolean map that is usually represented as graph, called interaction graph, where the vertices or nodes represent genes and the edges represent regulatory interactions. A gene in the network can be active or inactive and that is represented by a node in *state* 1 or 0 respectively. This model was first introduced by Kauffman in the end of the 60's [25] and it was thought as a generalization of the McCulloch and Pitts neural network model [28]. The seminal papers by Kauffman and Thomas focused in studying the dynamical properties of random generated networks [25,23,26] as well as studying the structure involved in the regulatory circuitry [34,35]. A boolean network naturally defines a discrete dynamical system by updating all the nodes of the network simultaneously, i.e. the consecutive states of the dynamics are given by iterations of the original boolean map. This update scheme is called *parallel* or *synchronous.* As the number of posible states is finite (it is given by the number of possible tuples with values 0 and 1 which is 2^n) every initial state eventually exhibit periodic dynamics. We call the set of states that define these periodic dynamics an *attractor.* If the attractor is one single state we call it a *fixed point* and otherwhise we call it a *limit cycle.* Though this model is fairly simple to study, it fails to reproduce gene expression data in a realistic way, mainly because of the synchronous update scheme. One straightforward approach to improve the model is adding asynchronicity to the dynamics by considering different update schemes [2,6,13,33]. Since some biologists agreed that some synchronicity is not completely unrealistic [5,37] updates schemes usually range from synchronous to sequential update schemes in which every node is updated according to a given partial order. A notable example of update schemes that are somewhere between the latter categories are *block sequential* update schemes. In these update schemes, a partition in the node set is defined and

nodes inside each set of the partition are updated in parallel while sets in the partition are updated sequentially. However, in order to define one of the latter update schemes, a partition and an partial order need to be chosen. These requirements introduce, in the biological networks modelling framework, several ways to model the dynamics of a fixed object of study. Although it is relevant and interesting from a mathematical or computational point of view to study the dynamics generated by every possible update scheme in the latter context, this exercise might turn to be rather impractical

An alternative approach to allow adding asynchronicity to the dynamics of a boolean network is based in the concept of delay that is generally defined as an internal clock, that could be independent from the original dynamics of the system, and that dictates its dynamical behaviour during a fixed time interval. This latter concept was first introduced by Thomas in [35,36] and then studied in different frameworks such as in [1,4,7,31,32]. Particularly, here we are interested in specific type of delay called *firing memory*. It was based in the concept of memory and it was first introduced by Graudenzi and Serra under the name of *gene protein Boolean networks* [16,18,17] and they defined this delay inspired in the concept of decay of proteins. In [9], Goles et al. introduced some modifications of the original model and presented it under the name of *Boolean networks with firing memory.* A question that naturally arise in this context is what are the effects of firing memory in the dynamics of boolean networks. According to [9], one of the first observations stated in the seminal papers, that was deduced through the analysis of numerical simulations, is that the higher the maximum time decay value (delay) the less the network admits asymptotic degrees of freedom. In order to survey this observation from a theoretical point of view, a straightforward methodology is to study a specific class of boolean networks preferably the one where the dynamics have been characterized. In fact, we are interested in the effect of *firing memory* in the dynamics of threshold networks. In these networks, the state of every node evolves accordingly to a *threshold function* that depends on the state of certain variables represented as the neighbors of the node in underlying interaction graph. In [15] Goles et. al characterized the dynamics of the latter network (without delay) and particularly they showed that attractors can only be limit cycles with period 2 or fixed points. One of the simplest type of threshold networks are the *disjunctive* and *conjunctive* boolean networks in which the state of every node depends on an OR or an AND function of its neighbors respectively. In [9], Goles et. al proved that disjunctive networks with firing memory only admit homogeneous fixed points as attractors. However, the effect of these type of delay in the dynamics of conjunctive networks have not been described until now. Perhaps surprisingly, conjunctive networks with firing memory can exhibit extremely different behaviour compared to the regular ones.

In the latter context, an interesting question is if firing memory is able to induce in the original boolean network dynamics the capability of simulating other computation models such as boolean circuits, Turing machines, etc. This line of research led us to consider a natural problem that arise in the study of boolean network dynamics: the *prediction problem.* This problem is defined in the following way: given an initial condition and an update scheme(in this case parallel scheme with firing memory), to predict the future states. To solve that problem, several strategies can be proposed from directly simulating the network to more elaborated strategies based on the topological or algebraical properties of the network. A measure of the efficiency of an strategy is given by the computational complexity of the problem. Prediction problems have been broadly studied in threshold networks [10,11] and particularly, in disjunctive(conjunctive) networks, it is known that the problem is in the **P** class.

In this paper, we focus in studying the dynamics of conjunctive networks with firing memory and we prove that, contrary to what might be assumed based on previous results for disjunctive networks, conjunctive networks with firing memory admit attractors of non polynomial period. Then, we study the prediction problem and we prove that it is **PSPACE**-complete. We achieve this by showing that conjunctive networks with firing memory are capable of simulating iterated boolean circuits.

As a direct corollary of this result, we conclude that the latter boolean networks with firing memory are universal, in the sense that they are able to simulate an arbitrary given boolean function.

2 Contributions and structure of the paper

In this paper we show that, contrarily to what one may think, conjunctive networks with firing memory exhibit an extremely complex dynamical behavior. More precisely, we show that 2-And-Prediction is **PSPACE**-complete as a consequence of the capability of this rule to simulate iterated monotone boolean circuits. As a corollary of the latter result, we show that conjunctive boolean networks with firing memory are a universal model in the sense that they are capable of simulating every boolean network automata.

The rest of the paper is organized as follows. In Section 3 we give the main formal definitions and previous results. In Section 4 we show the gadgets that play an essential role in the proof of our main results and we use them to exhibit a conjunctive network with firing memory that admits attractors of non polynomial period. In Section 5 we study the computational complexity of the 2-And-Prediction problem and we give a complete proof of the main result in Theorem 3.

3 Preliminaries

A boolean network is a map $F : \{0,1\}^n \to \{0,1\}^n$. Associated to this function, we define its interaction graph $\mathcal{G}(f) = (V, E)$ by $V = \{1, \ldots, n\}$ and $ij \in E \iff F_j$ depends on the variable x_i. F defines a dynamical system $(X = \{0,1\}^n, F)$ in which the elements $x \in X$ are called *states* or *configurations.* and the transitions are given by the iterations of the map F, i.e, for every state $x \in X$ we define its next state by $x(1) = F(x)$ and in general we have that $x(t+1) = F(x(t))$ for every $t \in \mathbb{N}$ $(x(0) = x)$. This type of dynamics is often called *parallel* or *synchronous* update scheme. In the next sections we will assume that boolean networks dynamics will be defined in this way. Given an initial condition $x \in X$, we call its associated *trajectory* to infinite sequence $T(x) = (x(0) = x, x(1), \ldots)$. As the number of possible states is finite (2^n), every trajectory is eventually periodic, i.e., there exists $p \geq 0$ such that $x(t+p) = x(t)$. We say that a trajectory reaches a *limit cycle* with period p if the last property hold for that trajectory and p is the minimum time in which the property is satisfied. A set of configurations in a limit cycle with period p is called an *attractor* with period p. Particularly, when $p = 1$ we say that the attractor is a *fixed point.*

We are interested in some specific type of boolean networks called *threshold* networks. A *threshold* network is a boolean network in which given a matrix $A = (a_{ij})$ with integer entries and an integer vector $\Theta = (\theta_i)_i$ the function F is defined by

$$F(x)_i = \begin{cases} 1 & \text{if } \sum_{j=1}^{n} a_{ij}x_j - \theta_i \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

One particular class of threshold networks are disjunctive and conjunctive networks. Disjunctive networks are defined by threshold 1 in every coordinate function F_i, in other words, are defined by an OR of certain variables i.e., $F(x)_i = F_i(x_{j_1}, \ldots, x_{j_k}) = \bigvee_{i=1}^{k} x_{j_i}$. On the other hand, a conjunctive network is a boolean network F such that every local rule F_i is given by an AND function of certain variables, i.e., $F(x)_i = F_i(x_{j_1}, \ldots, x_{j_k}) = \bigwedge_{i=1}^{k} x_{j_i}$. In this case, we have that $\theta_i = \delta_i$ for every i where δ_i is the number of neighbors of i in the interaction graph associated to F. These networks

have been broadly studied in different frameworks [12,22,3,8] mainly because their simplicity and their relevance in applications in modelling gene regulatory networks in which conjunctive functions describe common regulatory interactions [30,19].

As we referred in the introduction, the concept of delay in boolean networks has emerged as an alternative approach to introduce asynchronicity. In particular, we are interested in studying the effects of a type of delay called *firing memory*. We consider a boolean network F and states $Y = \prod_{i=1}^{n}\{0,1\} \times \{1,\dots,dt_i\}$, $dt_i \geq 1$ for all i. Given $y(0) \in X$, $y(0)_i = (x(0)_i, \Delta(0)_i)$, we define the following dynamics:

$$x(t+1)_i = \begin{cases} 1 & \Delta(t+1)_i \geq 1, \\ F_i(x(t)) & \Delta(t+1)_i = 0. \end{cases}$$

$$\Delta(t+1)_i = \begin{cases} dt_i & F_i(x(t)) = 1, \\ \Delta(t)_i - 1 & F_i(x(t)) = 0 \wedge \Delta(t)_i \geq 1, \\ 0 & F_i(x(t)) = 0 \wedge \Delta(t)_i = 0. \end{cases}$$

This local rule $(x(t)_i, \Delta(t)_i) \to (x(t+1)_i, \Delta(t+1)_i)$ defines a global transition function $F^{dt} : Y \to Y$ that we call boolean network with *firing memory*.

Remark 1 *One useful notation introduced in [9], is considering the states as the single delay value instead of a tuple. For example, the state* $(1,2)$ *that means state* 1 *and delay* 2 *is represented exclusively by* 2*. In the next sections, we will be using this notation.*

Remark 2 *We are implicitly assuming that a node* i *can not have simultaneously state* $x_i = 0$ *and delay* $\Delta_i \neq 0$*. In particular, if* $y = (x, \delta) \in Y$ *is an initial condition and* $\delta_i = 0$ *for some* $i \in V$ *then we must have* $x_i = 0$*.*

A natural question regarding the effects of firing memory in the dynamics of conjunctive networks is if this type of delay is able to give simulation capabilities to the network in the sense of allowing it to simulate other boolean networks of a different class or other computation models. In this context, we are interested in studying *prediction problems.* A well studied topic in the context of the dynamics of boolean networks is to make predictions about the attractor associated to an specific trajectory defined by a initial condition x. There exists a very simple solution to this problem that is to simulate the network dynamics until the initial state reaches a limit cycle. However, a question that naturally arise is if there exists a more efficient solution, considering the fact that using the last strategy may take as many steps as there are possible states. These more efficient solutions would be based on algorithmic or algebraic properties of the network. If $x \in \{0,1\}^n$ we introduce the complement of x denoted by $\bar{x}$ and defined by: $x_i = 1$ implies $\bar{x}_i = 0$ and $x_i = 0$ implies $\bar{x}_i = 1$. Given a maximum delay vector dt we define the following decision problem:

dt-AND-PREDICTION: Given a conjunctive network with firing memory F with maximun delay vector dt, $i \in \{1,\dots,n\}$ and a configuration $y = (x,\delta) \in Y$, does there exist $z = (w,\tau) \in T(y)$ such that $w_i = \bar{x}_i$?

We remark that, because we are working with the AND rule, if some configuration (x, δ) satisfies that $x_i = 0$ (that means that its delay value δ_i is considered to be 0) and $F(x)_i = 0$ then $x(t)_i = 0$ for all $t \geq 2$ (and so $\delta(t)_i = 0$), so the dt-AND-PREDICTION problem can be solved simulating one step of the local rule. Because of the latter observation, we can asume that $x_i = 1$. In addition, when we consider the maximum delay vector as uniform, i.e. the same maximum delay $dt_i = \tau$ in every i, we will refer to dt-AND-PREDICTION as τ-AND-PREDICTION. We are interested in studying the computational complexity of the previous problem. This concept is roughly defined as the amount of resources that are needed to find a solution, given as an expression of the input size. Classical theory defines the following main classes of complexity: **P** is the class of problems solvable by a

deterministic Turing machine in polynomial time and **PSPACE** is the class of problems solvable by a deterministic machine that uses polynomial space. Additionally, it is known that $\mathbf{P} \subset \mathbf{PSPACE}$. It is conjectured that these inclusions are strict, so there are problems in **PSPACE** that do not belong to **P**. The problems in **PSPACE** that are the most likely to not belong to **P** are the **PSPACE**-complete problems, which analogously **NP**-complete problems, are the ones such that any other problem in **PSPACE** can be reduced to them in polynomial time.

One very well known type of **PSPACE**-complete problem is related to the iterative evaluation of boolean circuits. A boolean circuit is a directed acyclic graph C that have three types of vertices: the ones with in-degree 0 called *inputs*, the ones with out-degree 0 called *outputs* and the rest of the vertices that have in and out neighbors called logical gates. These nodes are labelled by $\wedge, \vee, \neg$. A boolean circuit simulates a boolean function in the obvious way, and because of that, usually a circuit with n inputs and m outputs is denoted by $C : \{0,1\}^n \to \{0,1\}^m$. A circuit C is monotone if there are no gates labelled by $\neg$. For each gate of a circuit, its *layer* is the length of the shortest path from an input to the gate. A monotone circuit is alternating if for any path from an input to an output the gates on the path alternate between OR and AND gates. In addition, the inputs are connected to OR gates exclusively and outputs are OR gates. We define the following decision problem: Given a (monotone) boolean circuit $C : \{0,1\}^n \to \{0,1\}^n$, an input $x \in \{0,1\}^n$, and $i \in \{0, \ldots, n\}$ whether there exists a time $t \geq 1$ such that $C^t(x)_i = 1$. We call this problem Iter-Circuit-Prediction (respectively Iter-Mon-Circuit-Prediction).

Proposition 1 ([11]). Iter-Mon-Circuit-Prediction *is **PSPACE**-complete even when restricted to alternating circuits of degree 4.*

3.1 Previous results

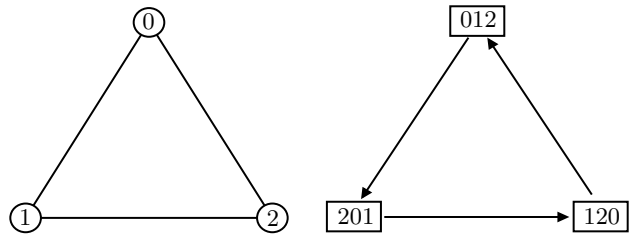
Threshold networks were vastly studied in [14,15,21,29] and Goles et. al [15] showed using a technique based on monotone energy operator that the *synchronous* dynamics associated to these networks admits only attractors of bounded period (moreover, there are only attractors with period 2 and fixed points) when the associated weight matrix is symmetric, i.e. when the underlying interaction graph is non-directed which is the case we are most interested on in this paper.

A wide studied subclass of threshold networks are the conjunctive or disjuntive networks. These systems have a very important role in modelling of biological systems because mainly because of its simplicity and their straightforward way to describe common interactions between different variables. The dynamics of these type of networks was studied under different update schemes in [12]. Our approach here is to continue the studies presented in the seminal paper of the *firing memory* [9] in which dynamics of disjunctive networks with firing memory were characterized. In the latter paper, Goles et. al showed that disjuntive networks with firing memory and delay $dt_i \geq 2$ in at least one coordinate i admit only homogeneous fixed points when the network is defined over a strongly connected directed graph.

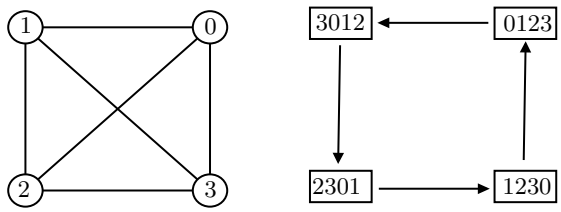
4 Conjunctive networks with firing memory admit non-polynomial cycles.

One surprising observation about the effect of firing memory in the dynamics of conjunctive networks is that it allows the dynamics to have attractors with period $p \geq 3$. We recall that these type of dynamics(without delay) have bounded cycles of maximum period $p = 2$ [15]. A general method to construct dynamics with a given period $p \geq 3$ is considering a conjunctive network defined by an interaction graph given by a complete graph K_{p+1} and a firing memory $dt_i = p$ in every vertex.

Proposition 2. *Let $\tau \geq 2$. There exists a conjunctive network with firing memory of size $\tau + 1$ and delay $dt_i = \tau$ in every i allowing attractors with period $p = \tau + 1$.*



(a) Conjunctive network with firing memory and delay $dt_i = 2$ in every node that admits a limit cycle with period 3. On the left hand side we show the interaction graph of the network and on the right hand side there is the transitions graph of the cycle.



(b) Conjunctive network with firing memory and delay $dt_i = 3$ in every node that admits a limit cycle with period 4. On the left hand side we show the interaction graph of the network and on the right hand side there is the transitions graph of the cycle.

Fig. 1: Attractors with period $p = \tau + 1$ for $\tau = 2$ and $\tau = 3$

Proof. Consider the conjunctive network given by the function $F(x)_i = \bigwedge_{j \neq i} x_j$, where $i \in \{0, \ldots, \tau\}$. We have that its interaction graph is given by the complete graph with $\tau + 1$ vertices $K_{\tau+1}$ (see Figure 1). We are going to exhibit an attractor X with period $p = \tau + 1$. Let $x_0 = (0123 \ldots \tau)$ be the initial condition. Observe that every $i \in V$ has initial delay i and every node $i \neq 0$ is in state 1. Because of how we defined F we have that in the next state every node $i \neq 0$ will be set to 0. However, the only node that will actually be in state 0 in its next state is $i = 1$, because every node $i \in V \setminus \{0, 1\}$ has delay $\Delta(0)_i \geq 2$. On the other hand, we have that for $i = 0$ every of its neighbours is in state 1 in the initial condition, so it will be updated as τ in the next iteration. Thus, $x_1 = (\tau 0123 \ldots \tau - 1)$. Now, we have that every node except $i = 1$ is in state 1 and every node $i \in i \in V \setminus \{1, 2\}$ has delay $\Delta_i(1) \geq 2$ so using the same argument we used for deducing x_1 we have that $x_2 = (\tau - 1\tau 0123 \ldots, \tau - 2)$. Iterating this process $\tau + 1$ times we verify we have a cycle with period $p = \tau + 1$:

$$X = \begin{cases} x_0 = (0123 \ldots \tau - 1\tau) \\ x_1 = (\tau 0123 \ldots \tau - 1) \\ x_2 = (\tau - 1\tau 0123 \ldots, \tau - 2) \\ \vdots \\ x_\tau = (123 \ldots \tau - 1\tau 0) \\ x_0 = (0123 \ldots \tau - 1\tau) \end{cases}$$

As a consequence of the last proposition, we have a stronger result on the period of the attractors when we consider conjunctive networks with firing memory with different maximum delay values.

Theorem 1. *There exists a connected conjunctive network of size n with firing memory (and not necessarily the same values for maximum delay) which admits attractors with period $2^{\Omega\left(\sqrt{n \log n}\right)}$.*

Proof. Let us consider a fixed natural number $m \geq 3$ and a collection of prime numbers $p_i, \ldots, p_l$, such that $2 \leq p_i \leq m$ where $\ell = \pi(m)$ and $\pi(m)$ denotes the number of primes not exceeding

m. For each $i \in \{1, \ldots, \ell\}$ we consider a conjunctive network with interaction graph given by a complete graph K_{p_i} as we did in the proof of the last proposition. We consider a graph G defined as the connected union of the previous complete graphs in the following way: we consider $V = \bigcup_{i=1}^{\ell} V(K_{p_i}) \cup \bigcup_{i=1}^{\ell-1} \{r_i\}$ and for every $i \in \{1, \ldots, \ell\}$ we choose an arbitrary vertex $s_i \in K_{p_i}$ and we consider $E = \bigcup_{i=1}^{\ell} E(K_{p_i}) \cup \bigcup_{i=1}^{\ell-1} \{s_i r_i\} \cup \bigcup_{i=1}^{\ell-1} \{s_{i+1} r_i\}$. In other words, we consider the union of the previous complete graphs and we connect each other by a unique 2-path.

Let us define the label function $\varphi : V \to \{1, \ldots, \ell\}$ given by $\varphi(u) = i$ if $u \in V(K_{p_i})$. Let $F^{p_1}, \ldots, F^{p_l}$ be the conjunctive networks with firing memory associated to each complete graph K_{p_i}. We recall that each of these functions has a maximum delay value of $p_i - 1$ in every node. We define the following conjunctive network with firing memory $F : \{0,1\}^{|V|} \to \{0,1\}^{|V|}$ given by $F(x)_i = F^{p_{\varphi(i)}}(x)_i$ for every $i \in V \setminus \{s_j | j \in \{1, \ldots, \ell\}\}$, $F(x)_{s_i} = F^{p_{\varphi(s_i)}}(x)_{s_i} \wedge x_{r_i} \wedge x_{r_{i+1}}$ for $i \in \{2, \ldots, \ell-1\}$, $F(x)_{s_\ell} = F^{p_{\varphi(s_\ell)}}(x)_{s_\ell} \wedge x_{r_{\ell-1}}$, $F(x)_{s_1} = F^{p_{\varphi(s_1)}}(x)_{s_1} \wedge x_{r_1}$ and $F(x)_{r_i} = x_{s_i} \wedge x_{s_{i+1}}$ for $i \in \{1, \ldots, \ell-1\}$. It is not difficult to see that the interaction graph of F is G. We note that functions $F(x)_{r_i}$ are regular conjunctive functions, in other words, delay values for this nodes are $dt_i = 1$.

Finally, let us define the initial condition $x \in \{0,1\}^{|V|}$ in the following way: for the vertices in $V(K_{p_i})$ we assign the initial condition $(0123, \ldots, p_i)$ and $x_{r_i} = 1$ for all $i \in 1, \ldots, \ell - 1$. Note that we cannot have two nodes connected and both having initial state 0 because of the nodes in different components are connected through the vertices r_i. Thus, the global dynamics converge to the fixed point 0. It is not difficult to see that starting from x every node in K_{p_i} is in a cycle with period p_i. Thus, we have that if T is the global period of the network then:

$$T \geq \prod_{i=1}^{\pi(m)} p_i.$$

And also, we have that:

$$|V| = \sum_{i=1}^{\pi(m)} (p_i + 1) - 1. \tag{1}$$

Additionally, if we define $\theta(m) = \sum_{i=1}^{\pi(m)} \log(p_i)$, we have that:

$$T \geq \exp(\theta(m)). \tag{2}$$

Based in 1, and 2, we are going to apply a technique used in [27] to deduce that T is not polynomial. This is based a result stated in [20]:

$$\pi(m) = \Theta\left(\frac{m}{\log(m)}\right), \tag{3}$$

$$\theta(m) = \Theta\left(\pi \log(m)\right). \tag{4}$$

We observe that from 3 it can be deduced that $m = O(\pi(m) log(m))$ and then $|V| = O(\pi(m)^2 \log(m))$. On the other hand, using this last observation we deduce that $\log|V| = O(\log(m))$. And finally, $\sqrt{|V|\log(|V|)} = O(\pi(m)\log(m))$ which is equivalent to say that $\pi(m)\log(m) = \Omega\left(\sqrt{|V|\log(|V|)}\right)$ and thus

$$T \geq \exp(\Omega\left(\sqrt{|V|\log(|V|)}\right).$$

We conclude that F has attractors with non polynomial period.

One natural question that arises in the context of the last theorem is if we can say something about the period of the attractors in the case when we restrict a conjunctive network with firing memory to have the same delay τ in all its coordinates. Would there exist a network of this class whose dynamics allows attractors with non-polynomial period? The answer is yes, but in order to exhibit it, we need to prove a proposition that is analogous to Proposition 2.

Proposition 3. *Let $\tau \geq 2$. For every integer $k \geq 2$, there exists a conjunctive network with firing memory of size $k\tau^2(\tau+1)$ and maximum delay $dt_i = \tau$ in every node i which admits attractors with period $k(\tau+1)$.*

Proof. Let $k \geq 1$ be an integer. Let us define $C = K_{\tau+1}$ as complete graph with $\tau+1$ vertices. We recall that these gadget defines a conjunctive network with firing memory which allows cycles of length $\tau+1$ when we have that the maximum delay of every node is $dt_i = \tau$. We are going to call this structure a clock. We are going to define k copies of a certain gadget that we will call block. Let $j \in \{0, \ldots, k\}$ we define the j-block B_j as a $\tau+1$-path such that every node has exactly $\tau-1$ neighbours and each one of these is in a different clock beside its neighbour in the path as it is shown in Figure 2. As every block contains an induced path then we can number the vertices in the path defining an initial and a terminal vertex. We write $C(B_j)_{r,l}$ for $r = 1, 2, 3, \ldots, \tau$ and $l = 1, 2, 3, \ldots, \tau-1$ to denote the clocks of the j-block that are associated to the r-th vertex of the path. Besides, as every node in the block has $\tau-1$ neighbours(each one in a different clock), given a node r in a block B_j we will call its neighbour in the clock $C(B_j)_{r,l}$ an *anchor* of r and we will denote it by $C(B_j)_{r,l}a$ or simply by a when the context is clear. We remark that the only vertex in a clock that is connected to exactly one node in the path is its anchor. The other nodes are only connected to vertices in the clock. Finally we consider the graph G as the connected union of

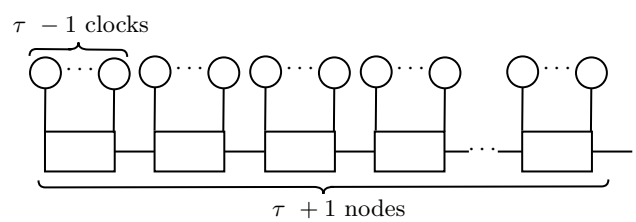


Fig. 2: Structure of the j-th block used in Proposition 2 to define a conjunctive network with firing memory and maximum delay values $dt_i = \tau$ for every node i that admits attractors with period $k(\tau+1)$. Every circle in the figure represents clock $C(B_j)_{r,l}$ associated to a node r represented by a square. This gadget has $\tau+1$ nodes and every node has $\tau-1$ clocks.

k-blocks defined connecting the terminal vertex of the j-th block to an initial vertex in $(j+1)$-th block and the terminal vertex of k-block to the initial vertex of the 1st-block. With this construction G is a $k(\tau+1)$-cycle in which every node is connected to a clock. Using the structure of G, we define a global rule F^τ as a conjunctive network with firing memory and maximum delay values τ which have as underlying interaction graph G.

Now, we define the dynamics of an attractor with period $k(\tau+1)$ using F^τ. We define the initial condition $x \in \{0,1\}^{|V|}$ by setting the same initial condition in each block except the first one. For first block we have the state $0123\cdots\tau$ for the nodes in the path and for $j = 2, 3, \cdots, k$ we have the initial states $\tau 123\cdots\tau$. For the clocks, we have that in each of these there is a unique anchor connected to one node in the path. We have by Proposition 2, that if we ignore this connection, every clock dynamics is associated to an attractor with period $\tau+1$. In fact, we will show that choosing a particular initial condition, every clock will exhibit this dynamics. We are going to write only the state of the anchor a in every clock and the other states in this subgraph are assumed to

be in the initial states so the clock defines an attractor with period $\tau + 1$. Assuming this notation, the initial state for the clocks are:

$$\underbrace{123\ldots(\tau-1)}_{\text{Node } 0}\ \underbrace{234\ldots\tau}_{\text{Node } 1}\ \underbrace{345\ldots\tau 0}_{\text{Node } 2}\ldots\ \underbrace{0123\ldots(\tau-2)}_{\text{Node } \tau}.$$

A summary of the initial condition x is shown in Figure 3.

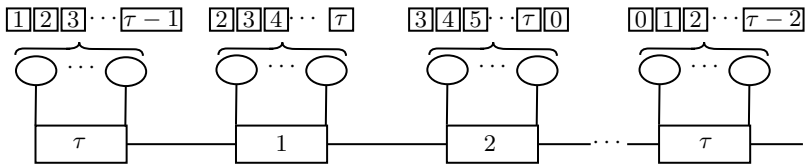


Fig. 3: Initial condition for the j-th block, $j \geq 2$ used in Proposition 2 to define an attractor with period $k(\tau + 1)$. For the first block we just define the state of the first node to 0 instead of τ.

Assuming this initial condition, we are going to show that the global period of the network is $k(\tau + 1)$. In order to that, it suffices to show that for every node i in the network such that $x_i \neq 0$, its next state will be given by the state of its neighbour that has state $x_i - 1$ and if $x_i = 0$ then its next state will be τ. In fact, if we have this property then, the next state of the first block will be $\tau 0123\ldots\tau - 1$ and if we continue iterating we will have that in $\tau + 1$ steps the second block will have the state $0123\cdots\tau$ and so in $k(\tau + 1)$ steps the network will return to the initial condition.

Let's see that our claim is true. Take the block B_j for some $j \geq 2$. We have that every node in the path, except for the first and second node, has some anchor a in state 0 as it is shown in Figure 4. Then, we will have that third to last nodes in the path will decrease their delay states by one and the first and second node will restore its delay to the maximum value because they have no neighbours in state 0. Thus, the next state for the nodes in the path will be $\tau\tau 123\ldots\tau - 1$. Besides, as anchors in clocks have no neighbours in the path with state 0 and other vertices in the clock are not connected to the path, clocks will be updated in the following way:

$$\underbrace{012\ldots(\tau-2)}_{\text{Node } 0}\ \underbrace{123\ldots\tau-1}_{\text{Node } 1}\ \underbrace{234\ldots\tau-1\tau}_{\text{Node } 3}\ldots\ \underbrace{\tau 012\ldots(\tau-2)}_{\text{Node } \tau}.$$

We note that now the first node in the path (that we call node 0) and its associated anchors have the same state than the last node in the path had the previous step, and the same hold for the other nodes in the block and in the clocks (they all assume the state of its left neighbour if they are in the path and every node in the clock decreases its delay by 1, except for the one with initial state 0 that restores its maximum delay τ). In the other hand, if we study the dynamics of the first block, we have the same behaviour than before except that because of node 0 is now initially setted to 0, then node 1 has a neighbour that is initially setted to 0 and so in the next step, every node in the path is going descend to the previous state except for the first one that will be setted to τ, In addition, all the clocks associated to the first block have the same behaviour than before except for the ones associated to the first node in the path. Nevertheless, because every anchor associated to the latter is in a state different than 0 and also because they all already have a neighbour in 0 inside the clock, they have the same transition than in the latter analysis. Thus, the next state of the nodes in the path will be $\tau 0123\ldots\tau - 1$ as desired and every one of these has now anchors in the same state than the previous state associated to its left neighbour. We conclude that F admits attractors with lenght $k(\tau + 1)$.

An immediate consequence of Proposition 3 is that we can exhibit a conjunctive network with firing memory (that have the same maximum delay values in every coordinate) which admits attractors

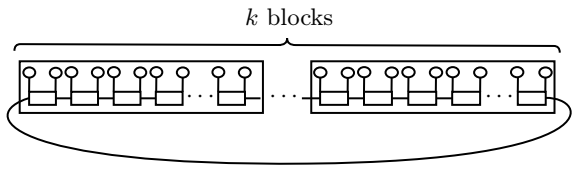


Fig. 4: Interaction graph G associated to a conjunctive network with firing memory and maximum delay $dt_i = \tau$ for all $i \in V(G)$, defined in Proposition 2 that admits attractors with length $k(\tau + 1)$.

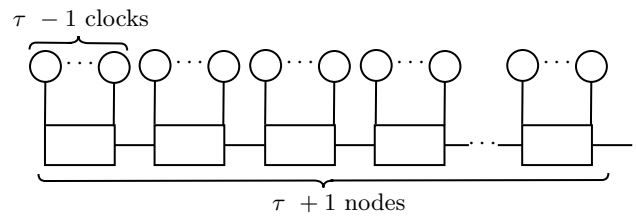


Fig. 5: Structure of a block used in Proposition 2 to define a conjunctive network with firing memory and maximum delay values $dt_i = \tau$ for every node i that admits attractors with period $k(\tau + 1)$. Every circle in the figure represents clock associated to a node r represented by a square. This gadget has $\tau + 1$ nodes and every node has $\tau - 1$ clocks.

with non polynomial period. This is possible because we can replicate, in this context, the same strategy we used to prove Theorem 1.

Theorem 2. *Let $\tau \geq 2$. There exists a conjunctive network of size n with firing memory and maximum delay $dt_i = \tau$ in every node i which admits attractors with period $2^{\Omega\left(\sqrt{n \log n}\right)}$.*

Proof. Let $m \geq 2$ and a collection of prime numbers $p_1, \ldots, p_l$ where $l = \pi(m)$ as in Theorem 1. As a consequence of Proposition 3 there exist functions F_{p_i} and associated graphs $C_{p_i},\ i = 1, 2, \cdots, l$ which admit attractors with period $p_i(\tau + 1)$. As same as we did to prove Theorem 1 we define a global function F with delay τ in every coordinate which associated interaction graph $G = (V, E)$ is given by the connected union of the graphs C_{p_i}. In this case, we connect every of these components by adding an edge between a node labelled by a associated to the first vertex in the path of the first block of C_{p_i} to another a labelled vertex associated to the second vertex in the path of the first block of $C_{p_{i+1}}$ as it is shown in Figure 6. Initial condition x is defined as every node in $C_{p_{i+1}}$ is in an attractor with period $(\tau + 1)p_i$ by using the same initial condition given in the proof of Proposition 3. We remark that there no connected vertices with state 0 in every iteration because of how we defined the connection between components. Again, as same as in Theorem 1 we have that the global period of the network T satisfy that:

$$T \geq (\tau + 1) \prod_{i=1}^{\pi(m)} p_i.$$

And also, we have that:

$$|V| = (1 + (\tau - 1)(\tau + 1))(\tau + 1) \sum_{i=1}^{\pi(m)} p_i. \tag{5}$$

It is not difficult to see that applying the same technique that we used in the proof of Theorem 1 we can conclude:

$$T \geq \exp(\Omega(\sqrt{|V| \log(|V|)}).$$

Thus, F has attractors with non polynomial period.

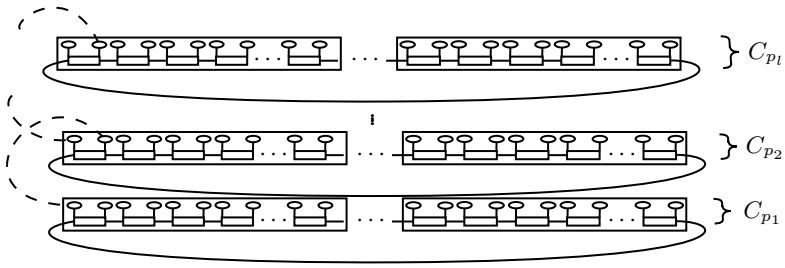


Fig. 6: Interaction graph G associated to a conjunctive network with firing memory and maximum delay values $dt_i = \tau$ for every node $i \in V(G)$ that admits attractors with non polynomial period. Every component defines a local dynamics with period $(\tau + 1)p_i$. Initial condition is defined verifying that there are no connected nodes in 0. Global period of the network is given by the product of prime numbers p_i.

5 2-And-Prediction is PSPACE-complete.

The fact that there exist conjunctive networks with firing memory that admit attractors with non polynomial period together with the structure of the gadgets we described, strongly suggest that there are conjunctive networks that are able to simulate boolean circuits. The next results confirm this insight establishing that conjunctive networks with firing memory are able to simulate iterated boolean circuits. We remark that previous results on the attractors period hold for an arbitrary value for the maximum delay that is defined for all the nodes in the given networks. So, we address the capability of conjunctive networks with firing memory and maximum delay $dt_i = 2$ in every node i to simulate an arbitrary iterated boolean circuit. These results have consequences related to computational complexity and universality of conjunctive networks with firing memory.

Proposition 4. *For every monotone boolean circuit $C : \{0,1\}^n \to \{0,1\}^n$ there is a conjunctive network F with firing memory such that: i) its interaction graph G has linear size in n, ii) its maximum delay values are $dt_i = 2$ in every node $i \in V(G)$ and iii) F simulates each iteration of C in linear time.*

Proof. Let C be a monotone circuit. We assume that: i) every input has out degree 1, ii) every output is identified with an input, iii) the degree of every logic gate in C is 4, and iv) every layer contains exclusively OR or AND gates and they are ordered alternately, i.e., if the k-th layer is made up of AND gates then the $k+1$-th is made up of OR gates (see Proposition 1). We represent this structure using the block gadget we defined for the last propositions (see Figure 5) with maximum delay vector $dt_i = 2$ in every node i. A scheme of the interaction graph associated to this conjunctive network with firing memory is shown in Figure 7. Let $i_1, \ldots, i_n$ be the inputs of C and because we are considering C as an iterated circuit, we are going to identify its outputs by the same names. For every $k \in \{1, \ldots, n\}$, we define a block B_{i_k}. These blocks are made up of a path of length three in which every node is connected to a clock. We introduce the following notation: if $x \in \{0,1\}^n$ and B is a block then $x_B = x_u x_v x_w$ where u, v and w are the labels of the vertices in the path. Given a initial configuration $x \in \{0,1\}^n$ for the circuit C, we code it using a variable $y \in \{0,1\}^m$, defined by maping the blocks in the following way:

$$y_{B_{i_k}} = \begin{cases} 122 \text{ if } x_{i_k} = 0 \\ 120 \text{ if } x_{i_k} = 1 \end{cases}$$

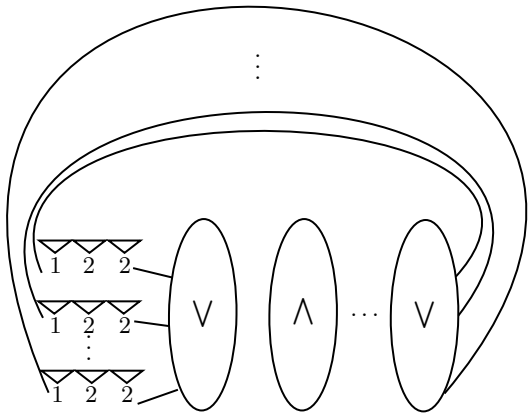


Fig. 7: Interaction graph associated to a conjunctive network with firing memory and maximum delay vector $dt_i = 2$ for every node i that simulates an iterated monotone boolean circuit C. Layers are made up by AND or OR gates exclusively, using the gadgets shown in Figure 8 and Figure 9 are alternately ordered.

For every logic gate AND or OR we define a gadget as the one showed in Figure 8 and Figure 9. Note that every logic gate is represented by a block so the last coding is well defined. We remark that as all the gadgets have a constant number of nodes and edges then, $m = O(n)$ and because of the assumptions we are doing on C we have that this coding has polynomial size in n. We will call φ the function such that $\varphi(x) = y$. Finally, as it is shown in Figures A.1 and Figure A.2 the information is transmitted through the blocks in a way such that in a maximum of 9 steps the nodes return to the initial condition so the circuit is cleared and the structures are available for continuing receiving and emitting signals. Then, we have given $x \in \{0,1\}^n$ and $y = \varphi(x)$, there exists $p \geq 1$ such that $\varphi(C^t(x))_{B_{i_k}} = (F^{pt}(y))_{B_{i_k}}$ for all $t \geq 0$ and $k \in \{1, \ldots, n\}$. Thus, the conjunctive network defined using these gadgets simulates C in linear space and linear time.

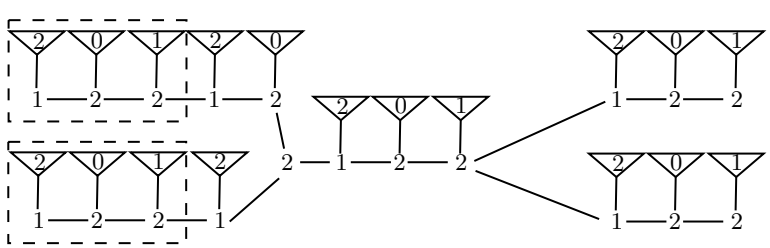


Fig. 8: Gadget of AND gates used in the graph shown in Figure 7. Signals are transmitted and coded based on the block gadget.

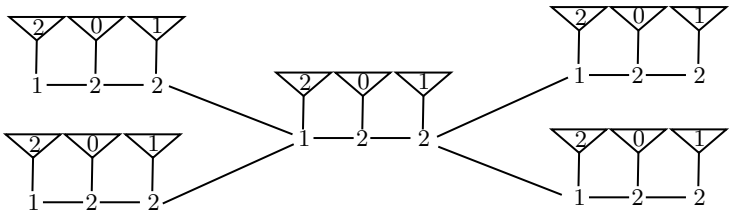


Fig. 9: Gadget of OR gates used in the graph shown in Figure 7. Signals are transmitted and coded based on the block gadget.

As a direct consequence of the Proposition 4, we have that conjunctive networks with firing memory are universal, i.e, they can simulate every boolean network. Finally, we address the question about the computational complexity of the prediction problem 2-And-Prediction. As a direct corollary of the latter proposition, we have that Iter-Mon-Circuit can be reduced to 2-And-Prediction and thus the problem is **PSPACE**-complete.

Theorem 3. *The problem* 2-And-Prediction *is* ***PSPACE****-complete.*

Proof. It is direct from Proposition 4 and Proposition 1

6 Discussion

In this paper, we have studied the effect of specific type of delay called firing memory in the dynamics of conjunctive boolean networks. More specifically, we have addressed the prediction problem in conjunctive networks with firing memory whose maximum delay is 2 in every node. We concluded that not only these type of networks admit attractors of non polynomial period but the latter problem turned out to be **PSPACE**-complete. Deducing this result was possible because of: i) the capability of conjunctive networks with firing memory that have the same value for maximum delay in every node, to have attractors with an arbitrary long period (Proposition 3) and ii) the capability of transmitting information through a wire that clears itself once the information has been transmitted (construction of blocks and clocks). These two main observations about conjunctive networks with firing memory allowed us to deduce the structure of the main gadgets used for the proof of our main results. These properties are quite surprising considering that conjunctive boolean networks admit only attractors of bounded period and the prediction problem is in **P**. Moreover, previous result on the effect of firing memory in the dynamics of the dual version of these type of networks, the disjunctive networks, suggested that firing memory tend to freeze the dynamics of these networks, reducing the period of the possible attractors that the network admits. We remark the relevance of the achieved results as they show that firing memory have effects on the dynamical properties of the original network that are similar to the ones exhibited by other update schemes that are somehow between synchronous and asynchronous dynamics in other type of boolean networks, such as the effects of block sequential update scheme in majority rules. In fact, in the latter case, the prediction problem with parallel update is **P**-complete while it is **PSPACE**-complete when we consider a block-sequential update scheme. It might be possible to deduce from the latter observation that firing memory allows to add asynchronicity to the dynamics of a boolean network in a less arbitrary way compared to block sequential update, that needs a predefined partition and an specific partial order.

An interesting topic for future work is the characterization of the dynamics of conjunctive networks with firing memory. While we have described conjunctive networks with firing memory that admit attractors with period proportional to the maximum delay values, the question about the existence

of networks admitting attractors with period not necessarily a multiple of their maximum delay values remains still open. In addition, the effect of firing memory in networks defined by particular topologies such as planar graphs or two dimensional grids might be interesting to analyse. Besides, considering the fact that in the light of the results of this paper there is no clear insight about a general effect of firing memory in a simple class of boolean functions such as AND or OR (we have complex dynamics in one case and we have dynamics that admit only fixed points in the other), an interesting topic for future work could be studying prediction problems in other classes of boolean networks with firing memory that are somehow similar to conjunctive networks (other functions that are linear for example) such as XOR networks.

7 Acknowledgement

This work has been partially supported by: CONICYT via PAI + Convocatoria Nacional Subvención a la Incorporación en la Academia Año 2017 + PAI77170068 (P.M.), CONICYT via PFCHA/DOCTORADO NACIONAL/2018 – 21180910 + PIA AFB 170001 (M.R.W) and ECOS C16E01 (E.G and M.R.W). We would also like to thank Alejandro Maass who provided insight and expertise that greatly assisted us in the course of this research. In addition, we would like to thank the anonymous reviewers for their helpful feedback and comments, particularly in the proofs of Proposition 2 and Proposition 3.

References

1. Ahmad, J., Roux, O., Bernot, G., Comet, J.P., Richard, A.: Analysing formal models of genetic regulatory networks with delays. International journal of bioinformatics research and applications **4**(3), 240–262 (2008)
2. Aracena, J., Goles, E., Moreira, A., Salinas, L.: On the robustness of update schedules in Boolean networks. Biosystems **97**(1), 1–8 (2009)
3. Aracena, J., Richard, A., Salinas, L.: Fixed points in conjunctive networks and maximal independent sets in graph contractions. Journal of Computer and System Sciences **88**, 145–163 (2017)
4. Bernot, G., Comet, J.P., Richard, A., Guespin, J.: Application of formal methods to biological regulatory networks: extending Thomas' asynchronous logical approach with temporal logic. Journal of theoretical biology **229**(3), 339–347 (2004)
5. Boettiger, A.N., Levine, M.: Synchronous and stochastic patterns of gene activation in the Drosophila embryo. Science **325**(5939), 471–473 (2009)
6. Demongeot, J., Elena, A., Sené, S.: Robustness in regulatory networks: a multi-disciplinary approach. Acta Biotheoretica **56**(1-2), 27–49 (2008)
7. Fromentin, J., Eveillard, D., Roux, O.: Hybrid modeling of biological networks: mixing temporal and qualitative biological properties. BMC systems biology **4**(1), 79 (2010)
8. Gao, Z., Chen, X., Başar, T.: Controllability of conjunctive Boolean networks with application to gene regulation. IEEE Transactions on Control of Network Systems **5**(2), 770–781 (2018)
9. Goles, E., Lobos, F., Ruz, G.A., Sené, S.: Attractor landscapes in Boolean networks with firing memory. Natural Computing (2019), accepted
10. Goles, E., Montealegre, P.: Computational complexity of threshold automata networks under different updating schemes. Theoretical Computer Science **559**, 3–19 (2014)
11. Goles, E., Montealegre, P., Salo, V., Törmä, I.: Pspace-completeness of majority automata networks. Theoretical Computer Science **609**, 118–128 (2016)
12. Goles, E., Noual, M.: Disjunctive networks and update schedules. Advances in Applied Mathematics **48**(5), 646–662 (2012)
13. Goles, E., Salinas, L.: Comparison between parallel and serial dynamics of Boolean networks. Theoretical Computer Science **396**(1-3), 247–253 (2008)
14. Goles-Chacc, E.: Comportement oscillatoire d'une famille d'automates cellulaires non uniformes. Ph.D. thesis, Institut National Polytechnique de Grenoble-INPG; Université Joseph-Fourier (1980)

15. Goles-Chacc, E., Fogelman-Soulié, F., Pellegrin, D.: Decreasing energy functions as a tool for studying threshold networks. Discrete Applied Mathematics **12**(3), 261–277 (1985)
16. Graudenzi, A., Serra, R.: A new model of genetic network: the Gene Protein Boolean network. In: Artificial Life and Evolutionary Computation, pp. 283–291. World Scientific (2010)
17. Graudenzi, A., Serra, R., Villani, M., Colacci, A., Kauffman, S.A.: Robustness analysis of a Boolean model of gene regulatory network with memory. Journal of Computational Biology **18**(4), 559–577 (2011)
18. Graudenzi, A., Serra, R., Villani, M., Damiani, C., Colacci, A., Kauffman, S.A.: Dynamical properties of a Boolean model of gene regulatory network with memory. Journal of Computational Biology **18**(10), 1291–1303 (2011)
19. Gummow, B.M., Scheys, J.O., Cancelli, V.R., Hammer, G.D.: Reciprocal regulation of a glucocorticoid receptor-steroidogenic factor-1 transcription complex on the Dax-1 promoter by glucocorticoids and adrenocorticotropic hormone in the adrenal cortex. Molecular Endocrinology **20**(11), 2711–2723 (2006)
20. Hardy, G.H., Wright, E.M.: An introduction to the theory of numbers. Oxford university press (1979)
21. Hopfield, J.J.: Neural networks and physical systems with emergent collective computational abilities. Proceedings of the national academy of sciences **79**(8), 2554–2558 (1982)
22. Jarrah, A.S., Laubenbacher, R., Veliz-Cuba, A.: The dynamics of conjunctive and disjunctive Boolean network models. Bulletin of Mathematical Biology **72**(6), 1425–1447 (2010)
23. Kauffman, S.: Homeostasis and differentiation in random genetic control networks. Nature **224**(5215), 177 (1969)
24. Kauffman, S.: The large scale structure and dynamics of gene control circuits: an ensemble approach. Journal of Theoretical Biology **44**(1), 167–190 (1974)
25. Kauffman, S.A.: Metabolic stability and epigenesis in randomly constructed genetic nets. Journal of theoretical biology **22**(3), 437–467 (1969)
26. Kaufmann, S.: Gene regulation networks: A theory of their global structure and behaviour. Current Topics in Developmental Biology **6**, 145–182 (1971)
27. Kiwi, M.A., Ndoundam, R., Tchuente, M., Goles-Chacc, E.: No polynomial bound for the period of the parallel chip firing game on graphs. Theoretical Computer Science. **136**(2), 527–532 (1994)
28. McCulloch, W.S., Pitts, W.: A logical calculus of the ideas immanent in nervous activity. The bulletin of mathematical biophysics **5**(4), 115–133 (1943)
29. Mortveit, H., Reidys, C.: An introduction to sequential dynamical systems. Springer Science & Business Media (2007)
30. Nguyen, D.H., D'haeseleer, P.: Deciphering principles of transcription regulation in eukaryotic genomes. Molecular systems biology **2**(1) (2006)
31. Ren, F., Cao, J.: Asymptotic and robust stability of genetic regulatory networks with time-varying delays. Neurocomputing **71**(4-6), 834–842 (2008)
32. Ribeiro, T., Magnin, M., Inoue, K., Sakama, C.: Learning delayed influences of biological systems. Frontiers in bioengineering and biotechnology **2**, 81 (2015)
33. Robert, F.: Discrete iterations: a metric study. Springer Science & Business Media
34. Thomas, R.: Boolean formalization of genetic control circuits. Journal of theoretical biology **42**(3), 563–585 (1973)
35. Thomas, R.: Regulatory networks seen as asynchronous automata: a logical description. Journal of theoretical biology **153**(1), 1–23 (1991)
36. Thomas, R., Thieffry, D., Kaufman, M.: Dynamical behaviour of biological regulatory networks—i. biological role of feedback loops and practical use of the concept of the loop-characteristic state. Bulletin of mathematical biology **57**(2), 247–276 (1995)
37. Wang, Q., Huang, J., Zhang, X., Wu, B., Liu, X., Shen, Z.: The spatial association of gene expression evolves from synchrony to asynchrony and stochasticity with age. PloS one **6**(9), e24076 (2011)

A Appendix

A.1 Supplementary figures: dynamics of logic gates gadgets

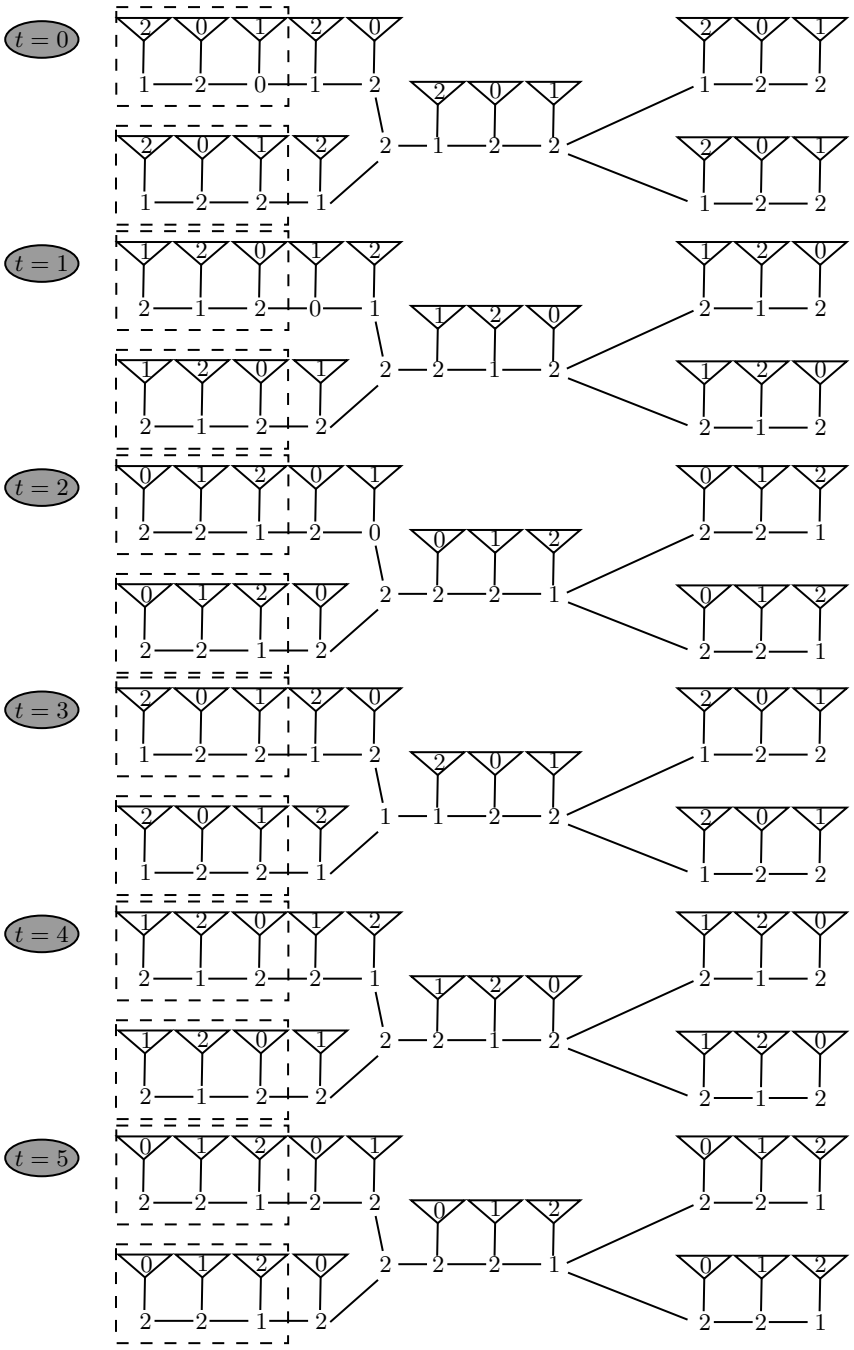


Fig. A.1: Iterations of the AND gate gadget. A 0 and a 1 are computed as inputs. After seven steps the information is transmitted and initial condition is recovered.

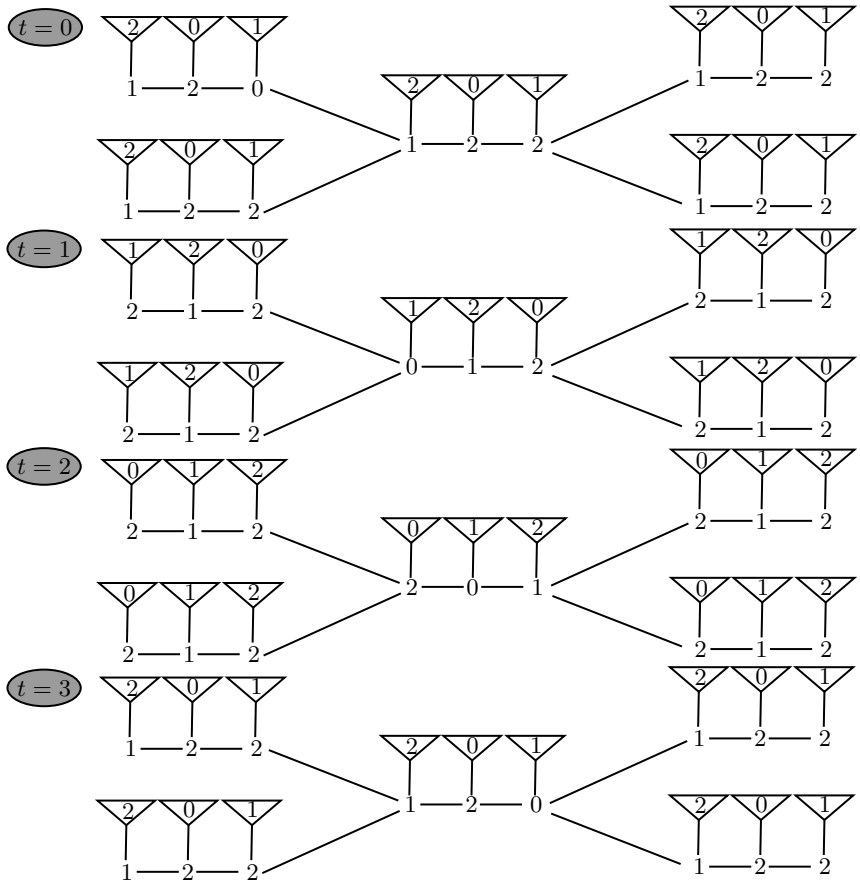


Fig. A.2: Iterations of the OR gate gadget. A 0 and a 1 are computed as inputs. After three steps the information is transmitted and the initial condition is recovered.