



HAL
open science

Synchronism versus asynchronism in monotonic Boolean automata networks

Mathilde Noual, Sylvain Sené

► **To cite this version:**

Mathilde Noual, Sylvain Sené. Synchronism versus asynchronism in monotonic Boolean automata networks. *Natural Computing*, 2018, 17 (2), pp.393-402. hal-01479438

HAL Id: hal-01479438

<https://hal.science/hal-01479438v1>

Submitted on 4 May 2018

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.

Synchronism vs asynchronism in monotonic Boolean automata networks

Mathilde Noual¹ and Sylvain Sené^{2,3}

¹Freie Universität Berlin, FB Mathematik und Informatik, Berlin,
Germany

²Université d'Aix-Marseille, CNRS, LIF, Marseille, France

³Institut rhône-alpin des systèmes complexes, IXXI, Lyon, France

1st March 2017

Abstract

This paper focuses on Boolean automata networks and the updatings of automata states in these networks. More specifically, we study how synchronous updates impact on the global behaviour of a network. On this basis, we define different types of network sensitivity to synchronism, which are effectively satisfied by some networks. We also relate this synchronism-sensitivity to some properties of the structure of networks and to their underlying mechanisms.

Key words: Boolean automata networks; sensitivity to synchronism; asynchronous transition graph; elementary transition graph; time in interaction systems

1 Introduction

Informally, automata networks are networks of interacting entities, the automata, whose discrete states evolve locally over discrete time. The origins of automata networks can be traced back to the introduction of formal neural networks [19] and cellular automata [32]. Later, in the 1980's, a thriving line of researches significantly contributed to sustaining interest in these objects [5, 12, 24]. In particular, Robert raised the problem of organising in time the updatings of automata states in his seminal book [24]. The more general definition of (finite) automata networks that is used today owes to these theoretical researches as well as to a biological lineage that applied automata networks to the modelling of genetic regulation networks [16, 28].

This dual legacy results in two very close standpoints that can be taken to consider automata networks and to motivate their study: the first one is computation-oriented, the second one is modelling-oriented. Notice that

these two standpoints give reasons to impose states of automata to take Boolean values. Indeed, on the one side, *Boolean automata networks* (BANs) have been proven to be Turing-complete, when they are of infinite size (*i.e.*, composed of an infinite number of automata). A nice proof of this result is presented in [12]. As a consequence, our decision to work on BANs is not a restriction from this point of view because, even in the finite case, any discrete interaction network can be simulated by a BAN of larger size. On the other side, their ability to model genetic regulation networks, highlighted from the end of the 1960's, comes notably from the intrinsic Boolean nature of genes, that can be expressed or not, depending on whether they are transcribed or not in messenger RNAs.

Beyond this classical view that comes from a molecular biology approach, let us add that our choice to study BANs rather than multi-valued automata networks is wilful. It directly comes from the first elementary problems that arise from our chosen stance which are more concerned by (the possibilities of) changes than the nature of changes. Clearly, we are more interested in the ability and the effectivity of any automaton to change state than in the fact that it changes state from a value v_1 to a new value v_2 .

In this paper, in lines of Robert, we address the question of how much impact the organisation in time of automata updates in monotonic BANs has on their (asymptotic) dynamics. In particular, basing ourselves on the classical non-deterministic asynchronous updating mode, the main contributions of this paper consists of:

- showing that adding synchronism, even a very little quantity, can drastically change the dynamical behaviour of monotonic BANs (this part is based on [20]);
- showing conditions on the static features of monotonic BANs that are necessary for them to be *synchronism-sensitive*.

In Section 2, we develop the two distinct standpoints mentioned above, which participate in clarifying our motivations and in presenting some important literature elements. Then, Section 3 introduces all the useful definitions, notations and preliminary remarks. It is followed by Section 4 that presents the results obtained highlighting the influence of updating modes on monotonic BANs: we first classify the different impacts that the addition of one synchronous transition may have on monotonic BAN behaviours, and we introduce the different sorts of possible synchronism-sensitivities and relate them to some structural properties. The paper ends with a discussion that both concludes on the work realised and opens on some of its perspectives, notably on local non-monotonicity.

2 Standpoints, motivations and state of the art

2.1 The computational standpoint

In a BAN, an automaton is a computing unit that takes as its input the current states of its neighbours (*i.e.*, automata that act on it) and that gives as its output its own new updated state. An interesting fact is that BANs originate finite automata, introduced by Kleene in [17] on the basis of [19]. However, it is crucial to understand that an automaton does not represent the same mathematical object according to the context. Indeed, in our context, the view on BANs has made considering the network entities themselves as automata and, for a given finite BAN, it is the global evolution of its entities over time (*i.e.*, the global dynamics of the network) that is a finite automaton. This allows to put emphasis precisely on the *network* of automata and on the computation that is performed globally by it and, consequently, to relate some of its static / syntactic properties to its dynamical / semantical ones. Thus, BANs are well suited to explore some of the fundamental mechanisms that computation involves, especially mechanisms that can be expressed in terms of properties of the underlying network structures [13]. They also particularly facilitate putting focus specifically on the notion of *change*.

BANs are closely related to cellular automata. The main and most current differences are that:

- the set of entities in a BAN is assumed to be finite (even if nothing prevents studying infinite BANs) and
- no regularity is assumed in the set of transition rules assigned to its entities, which induces notably that the structure of a BAN is not necessarily a lattice but a digraph.

An interesting point is that (a)synchronism has already been the object of some probabilistic studies in the context of cellular automata [4, 10, 18, 26]. However, in this context, *perfect synchronism*, classically called the parallel updating, is considered as the reference to which a certain rate of asynchronism is added. In this paper, on the contrary, we consider *perfect asynchronism* as the reference and propose to determine the possible effects of a punctual addition of synchronism. Our contribution to researches on (a)synchronism in cellular automata thus follows from these differences and from the combinatorial approach that we propose to adopt to start exploring the minimal, fundamental differences between synchronism and asynchronism.

Here, our ultimate aim is to contribute to the understanding of how the precise scheduling of events (local changes) in a system impacts on its possibilities. As a preliminary exploration in these general lines, this paper

proposes to start by determining whether synchronism – per se – in events may impact on the global computation that is made by the system, and, if it does, in what ways.

2.2 The modelling of interaction systems standpoint

BANs can also be seen as conceptual prototypes of interaction systems. Here, we use the terms “interaction system” to refer to any system that can potentially be defined as a set of interacting entities such that all events that are possible in these systems (*i*) are caused exclusively by interactions between entities of the system and (*ii*) correspond to state changes of these entities. In a BAN, the entities / automata are predetermined: when updated, their response to a given situation is invariable and predefined by their transition rule. Although BANs are typically used to model neural and genetic networks [3, 11, 12, 16, 19, 28], that is, interaction systems involving entities and interactions of the same kind, generally, in one same BAN, different automata may be made to represent entities of very different kinds and statuses, and their state values may be assigned very different meanings.

In these lines, let us highlight that since BANs are used to being regarded as models of specific biological systems, they are often assigned applicative purposes and scopes (more or less implicitly and distantly) whose intricacies often require to bypass theoretical complexity issues related to their definition. In particular, the additional complexity brought by the different possible ways of updating automata states has been dealt with in the past by imposing restrictions on the updating mode (*e.g.* restrictions to the non-deterministic asynchronous one [22, 23, 27, 29], the parallel one, or some deterministic [1, 2, 8, 14] or probabilistic ones [9, 15, 25]).

Here, we make no prior restriction on the way automata states are updated. Furthermore, instead of using the theoretical objects as tools to address applicative problems, we propose on the contrary to take a fundamental approach on the applicative objects that are covered under the terms “interaction systems”.

2.3 Synchronism

Importantly, let us also highlight that the notion of synchronism that we propose to consider here is very distinct from that of *simultaneity* (whose full meaning relies intrinsically on the notion of duration which is difficult to account for consistently in such a discrete model). In the past, the confusion between these two notions has often caused synchronism to be disregarded altogether on the basis of great unlikeliness of simultaneity in real systems. However, a synchronous transition of a BAN simply corresponds

to the possibility of several events occurring rather than just one. It can therefore most naturally be taken as the representation of the case where several events occur close enough in time so as to disallow an intermediary step in which automata could communicate, *i.e.* could become acquainted with the actual network configuration and adapt their response to it. For this reason, synchronism is an important notion essentially relating to the modelling of time in the theoretical framework of BANs. Understanding the impact of synchronism on BAN behaviours can thus possibly contribute to the understanding of the specific role and influence of time (more precisely of the organisation of events in time resulting from their relative durations and speeds) in interaction systems.

3 Definitions, notations and preliminary remarks

3.1 Automata, states and configurations

By default, $\mathbf{V} = \{0, \dots, n-1\}$ denotes a set of $n \in \mathbb{N}$ Boolean *automata*. We let $\mathbb{B} = \{0, 1\}$. A *configuration* of \mathbf{V} , or of a network \mathcal{N} with automata set \mathbf{V} , is a Boolean vector $x \in \mathbb{B}^n$ whose component $x_i \in \mathbb{B}$ represents the *state of automaton* $i \in \mathbf{V}$. In this paper, special attention is paid to switches of automata states starting in a given configuration, so we introduce the following notations:

$$\forall x = x_0 \dots x_{n-1} \in \mathbb{B}^n, \forall i \in \mathbf{V}, \quad \bar{x}^i = x_0 \dots x_{i-1} \neg x_i x_{i+1} \dots x_{n-1}$$

and

$$\forall W = W' \uplus \{i\} \subseteq \mathbf{V}, \quad \bar{x}^W = \overline{(\bar{x}^i)^{W'}} = \overline{(\bar{x}^{W'})^i}.$$

To switch from Boolean values to signed values, we let $\mathbf{s} : b \in \mathbb{B} \mapsto b - \neg b \in \{-1, 1\}$. Conversely, to switch from signed values to Boolean ones, we use $\mathbf{b} : a \in \{-1, 1\} \mapsto (a + 1)/2 \in \mathbb{B}$. Also, to compare two configurations $x, y \in \mathbb{B}^n$, we use $D(x, y) = \{i \in \mathbf{V} \mid x_i \neq y_i\}$ and the Hamming distance $d(x, y) = |D(x, y)|$.

3.2 Networks and mechanisms

A *Boolean automata network* (BAN) \mathcal{N} of size $n \in \mathbb{N}$ is a set of n Boolean functions (*cf.* Figure 1 a):

$$\mathcal{N} = \{f_i : \mathbb{B}^n \rightarrow \mathbb{B} \mid i \in \mathbf{V}\}.$$

f_i is the *transition rule* of, or *mechanism* executed by, $i \in \mathbf{V}$. It predetermines the possible behaviour of automaton i in each configuration $x \in \mathbb{B}^n$. More precisely, if i is *updated* in configuration $x \in \mathbb{B}^n$, then its state switches from x_i to $f_i(x)$.

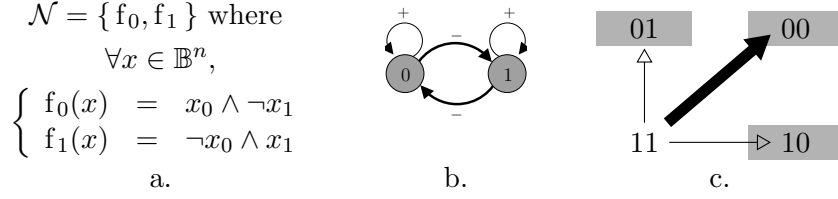


Figure 1: a. A BAN of size 2. b. Its signed structure (arcs (j, i) are labelled by $\text{sign}(j, i)$). c. Its ETG \mathcal{T} which only differs from its ATG \mathcal{T}_a by $11 \rightarrow 00$. This special case of F-impact is induced by a 11-critical, Hamiltonian cycle. Its impact, precisely, consists in making reachable an unreachable attractor of \mathcal{T}_a (cf. Lemmas 3 and 4).

3.3 Structure and influences between automata

We introduce the sign of the influence of $j \in \mathbf{V}$ on $i \in \mathbf{V}$ in $x \in \mathbb{B}^n$:

$$\text{sign}_x(j, i) = \frac{f_i(x) - f_i(\bar{x}^j)}{x_j - \bar{x}_j} = \mathbf{s}(x_j) \cdot (f_i(x) - f_i(\bar{x}^j)).$$

$\mathbf{A}(x) = \{(j, i) \in \mathbf{V}^2 \mid \text{sign}_x(j, i) \neq 0\}$ represents the set of influences of \mathcal{N} that are *effective* in x , which formally means that

$$\forall i, j \in \mathbf{V}, \exists x \in \mathbb{B}^n, f_i(x) \neq f_i(\bar{x}^j) \Leftrightarrow (j, i) \in \mathbf{A}.$$

The *structure* (cf. Figure 1 b) of \mathcal{N} is $\mathbf{G} = (\mathbf{V}, \mathbf{A})$, where $\mathbf{A} = \bigcup_x \mathbf{A}(x)$ represents the set of influences of \mathcal{N} . $\mathbf{V}^-(i)$ denotes the in-neighbourhood of $i \in \mathbf{V}$ in \mathbf{G} and $\text{deg}^-(i) = |\mathbf{V}^-(i)|$. Furthermore, let us add that, with no loss of generality, BANs considered in this paper are strongly connected (otherwise, each strongly connected component – SCC – is studied independently).

3.4 Monotonicity and signs

If $\forall x \in \mathbb{B}^n$, $\text{sign}_x(j, i)$ is constant when it is non-null, which is the case when $(j, i) \in \mathbf{A}(x)$, then we define its non-null value to be $\text{sign}(j, i)$. In particular, for the only arc (j, i) incoming i such that $\text{deg}^-(i) = 1$, $\text{sign}(j, i)$ is defined. Generally, if an arc (j, i) is signed and positive (*i.e.* if $\text{sign}(j, i) = +1$), then the state of i tends to mimic that of j . If it is negative ($\text{sign}(j, i) = -1$), it tends to negate it.

Here we assume that all arcs can be signed, which is equivalent to assuming all f_i s to be locally monotonic in all components. Non-monotonicity is a subject that we believe to be important in itself and that we have started to study in [20, 21]. Although we focus only on locally monotonic BANs in this paper, we mention local non-monotonicity again in the conclusion.

Note also that the *sign* of a path in \mathbf{G} is the product of the signs of its arcs. Thus, a positive path globally transmits “information” directly whereas a negative one transmits its negation.

3.5 Instabilities and frustrations

An automaton $i \in \mathbf{V}$ is *unstable* in $x \in \mathbb{B}^n$ if it belongs to the set

$$\mathbf{U}(x) = \{i \in \mathbf{V} \mid f_i(x) \neq x_i\}.$$

It is *stable* in x if it belongs to $\overline{\mathbf{U}}(x) = \mathbf{V} \setminus \mathbf{U}(x)$. *Stable configurations* $x \in \mathbb{B}^n$ are such that $|\mathbf{U}(x)| = 0$. Lemma 1 below relates the concept of instability with that of loops in interaction graphs.

Lemma 1 (loops). *For any automaton $i \in \mathbf{V}$, we have:*

$$\begin{aligned} (i, i) \in \mathbf{A} \text{ and } \text{sign}(i, i) = +1 &\Leftrightarrow \exists x \in \mathbb{B}^n, i \in \overline{\mathbf{U}}(x) \cap \overline{\mathbf{U}}(\bar{x}^i), \\ (i, i) \in \mathbf{A} \text{ and } \text{sign}(i, i) = -1 &\Leftrightarrow \exists x \in \mathbb{B}^n, i \in \mathbf{U}(x) \cap \mathbf{U}(\bar{x}^i). \end{aligned}$$

Proof. By the effectiveness of influences of BAN structures and the definition of signed arcs, (i, i) is a positive (resp. negative) loop if and only if $\exists x \in \mathbb{B}^n, x_i = 1$ such that $1 = x_i = f_i(x) > f_i(\bar{x}^i) = \bar{x}_i^i = 0$ (resp. $0 = \bar{x}_i^i = f_i(x) < f_i(\bar{x}^i) = x_i = 1$). \square

As we can see, the concept of stability applies to automata being given a configuration. A natural idea is to have a similar concept for influences. Thus, from [6, 30, 31], we derive the notion of a frustration for an influence. Intuitively, given a configuration x , an influence (j, i) is said to be frustrated if its isolated action on x would transform x in another configuration $y \neq x$. More formally, an influence $(j, i) \in \mathbf{A}$ is *frustrated* in a configuration $x \in \mathbb{B}^n$ if and only if it belongs to

$$\mathbf{FRUS}(x) = \{(j, i) \in \mathbf{A} \mid \mathbf{s}(x_j) \cdot \mathbf{s}(x_i) = -\text{sign}(j, i)\}.$$

Notably, it can be checked that $i \in \mathbf{U}(x) \Rightarrow \exists j \in \mathbf{V}^-(i), (j, i) \in \mathbf{FRUS}(x)$. For instance, if we consider a BAN of size one. Let i be this automaton such that it admits a negative loop. Whatever the configuration of this BAN, the loop (i, i) is always frustrated, which is also true for any negative loop in any BAN. Lemma 2 below highlights an interesting relation between frustrations and instabilities. The proof mainly relies on the local monotonicity of the mechanisms.

Lemma 2 (adding frustrations). *Adding frustrated influences incoming an unstable automaton cannot stabilise it. Formally, denoting $\mathbf{V}^-(i) \cap \{j \in \mathbf{V} \mid (j, i) \in \mathbf{FRUS}(x)\}$ by $\mathbf{V}_{\mathbf{FRUS}(x)}^-(i)$, we have:*

$$\forall x, y \in \mathbb{B}^n, i \in \mathbf{U}(x) \wedge \left(\mathbf{V}_{\mathbf{FRUS}(x)}^-(i) \subseteq \mathbf{V}_{\mathbf{FRUS}(y)}^-(i) \right) \Rightarrow i \in \mathbf{U}(y).$$

Proof. By hypothesis of local monotonicity and because the input provided by j to i is

$$\mathbf{b}_j^i(x) = \mathbf{b}(\text{sign}(j, i) \cdot \mathbf{s}(x_j)) = \begin{cases} x_i & \text{if } (j, i) \notin \mathbf{FRUS}(x) \\ \neg x_i & \text{otherwise} \end{cases},$$

any transition rule f_i can be written in conjunctive normal form as follows:

$$f_i : x \in \mathbb{B}^n \mapsto \bigwedge_{k < m} c_k(x),$$

where each of the m disjunctive clauses $c_k(x)$ involves a certain subset $\mathbf{V}_k^i \subseteq \mathbf{V}^{-i}$ of the in-neighbours of i so that $\mathbf{V}^{-i} = \cup_{k < m} \mathbf{V}_k^i$, and:

$$c_k(x) = \bigvee_{j \in \mathbf{V}_k^i} \mathbf{b}_j^i(x) = \bigvee_{\substack{j \in \mathbf{V}_k^i \\ (j, i) \in \mathbf{FRUS}(x)}} \neg x_i \quad \vee \quad \bigvee_{\substack{j \in \mathbf{V}_k^i \\ (j, i) \notin \mathbf{FRUS}(x)}} x_i.$$

Let x be an unstable configuration. By definition, x admits at least one frustrated incoming influence (otherwise, it could not be unstable). Let $y \in \mathbb{B}^n$ such that it admits at least one more frustrated incoming influence than x . Since f_i can be written as a conjunction of disjunctive clauses, it is easy to check that the value of these clauses for y are necessarily the same as for x . As a consequence, adding frustrated influences incoming an unstable automaton cannot stabilise it. \square

3.6 Transitions and transition graphs

An *elementary transition* of a BAN \mathcal{N} is a couple of configurations $(x, y) \in \mathbb{B}^n \times \mathbb{B}^n$, noted $x \rightarrow y$, satisfying $\mathbf{D}(x, y) \neq \emptyset \subseteq \mathbf{U}(x)$. It represents a *punctual* event of \mathcal{N} , defined by one or several state changes of its automata. Let $\mathbf{T} = \{x \rightarrow y \mid x, y \in \mathbb{B}^n\}$ be the set of elementary transitions of \mathcal{N} . In this framework, the *punctual events* experienced by a BAN \mathcal{N} correspond precisely to punctual state changes of one or several of its automata. Thus, \mathbf{T} represents the set of all punctual events that \mathcal{N} may be submitted to in each of its configurations. The digraph $\mathcal{T} = (\mathbb{B}^n, \mathbf{T})$ is called its *elementary transition graph* (ETG). It describes the *behaviour* of \mathcal{N} , that is, all of its possible evolutions (e.g. Figure 1 c).

The *size of an elementary transition* $x \rightarrow y$ equals $d(x, y)$. If $d(x, y) = 1$ (resp. $d(x, y) > 1$), then $x \rightarrow y$ is called *asynchronous* (resp. *synchronous*) and written $x \rightarrow\triangleright y$ (resp. $x \rightarrow\blacktriangleright y$). The set of asynchronous transitions is noted \mathbf{T}_a . The digraph $\mathcal{T}_a = (\mathbb{B}^n, \mathbf{T}_a)$ is called the *asynchronous transition graph* (ATG). It represents only those events that \mathcal{N} can undergo which involve only one local state change at a time. The transitive closures of \rightarrow and $\rightarrow\triangleright$ are denoted by $\rightarrow\gg$ and $\rightarrow\triangleright\gg$. Paths in a transition graph are called *derivations* and, abusing language, we speak of a *derivation* $x \rightarrow\gg y$ or $x \rightarrow\triangleright\gg y$.

4 Main results

4.1 Direct derivations and critical NOPE-cycles

A *cycle* of a BAN \mathcal{N} is a closed directed path of its structure \mathbf{G} . Before we go further, let us say that the study that follows leads us to pay particular attention to specific cycles which are going to be proven necessary for BANs to be sensitive to the adding of synchronism. We decided to call these specific cycles *critical cycles*, whose definition follows. $\forall x \in \mathbb{B}^n$, we say that the cycle $\mathbf{C} = (\mathbf{V}_{\mathbf{C}}, \mathbf{A}_{\mathbf{C}})$ is *x-critical* if $\mathbf{A}_{\mathbf{C}} \subseteq \mathbf{FRUS}(x)$, which implies that $\mathbf{V}_{\mathbf{C}} \subseteq \mathbf{U}(x)$ (cf. Section 3.5) and it is *critical* if it is *x-critical* for some $x \in \mathbb{B}^n$ (cf. Figure 1).

By definition of frustrated arcs, if $\mathbf{C} = (\mathbf{V}_{\mathbf{C}}, \mathbf{A}_{\mathbf{C}})$ is *x-critical*, has length ℓ and sign \mathbf{s} then $\prod_{(j,i) \in \mathbf{A}_{\mathbf{C}}} -\text{sign}(j, i) = (-1)^\ell \times \mathbf{s} = \prod_{(j,i) \in \mathbf{A}_{\mathbf{C}}} \mathbf{s}(x_j) \cdot \mathbf{s}(x_i) = 1$. This yields Proposition 1 below.

Proposition 1. *A cycle that is critical is a NOPE-cycle, i.e. negative with an odd length or positive with an even length.*

We will also make specific use of the concepts of inclined and unwilling configurations towards a given destination. The idea is that a configuration x is inclined (resp. unwilling) towards another one y if it admits (resp. it does not admit) an unstable automaton whose updating leads to a configuration x' closer to y , in terms of Hamming distance. More formally, for all configurations $x, y \in \mathbb{B}^n$ such that $x \neq y$, we say that x is *inclined* (resp. *unwilling*) towards y if $\mathbf{D}(x, y) \subseteq \mathbf{U}(x)$ (resp. $\mathbf{D}(x, y) \cap \mathbf{U}(x) = \emptyset$). Also, let $x = x(0) \rightarrow x(1) \rightarrow \dots \rightarrow x(m-1) \rightarrow y = x(m)$ be a derivation from x to y . If $\forall t < m$, $\mathbf{D}(x(t+1), y) \subsetneq \mathbf{D}(x(t), y)$, this derivation is said to be *direct*. It performs no *reversed changes*, i.e. $\forall t < m$, $x(t)_i = y_i \Rightarrow \forall t' < t \leq m$, $x(t')_i = y_i$.

Proposition 2. *Let $x \in \mathbb{B}^n$ be a configuration that is inclined towards configuration $y \in \mathbb{B}^n$. If there are no asynchronous derivations from x to y then $\mathbf{D}(x, y)$ induces a NOPE-cycle that is *x-critical*. If $\mathbf{D}(x, y)$ does not induce an *x-critical* cycle, then there is a direct asynchronous transition from x to y .*

Proof. Let us consider the digraph $\mathbf{H} = (\mathbf{D}(x, y), \mathbf{FRUS}(x) \cap (\mathbf{D}(x, y) \times \mathbf{D}(x, y)))$. Let m be the number of SCCs of \mathbf{H} and $\delta : \mathbf{D}(x, y) \rightarrow \{0, \dots, m-1\}$ be a structural ordering of the vertices of \mathbf{H} defined as:

$$\forall j, i \in \mathbf{D}(x, y),$$

$$(j, i) \in \mathbf{FRUS}(x) \Rightarrow \begin{cases} \delta(i) < \delta(j) & \text{if } j \text{ and } i \text{ do not belong to} \\ & \text{the same cycle in } \mathbf{H}, \\ \delta(i) = \delta(j) & \text{otherwise.} \end{cases}$$

Notice that, according to Lemma 2, increasing the number of frustrated arcs incoming an unstable automaton cannot make this automaton stable. On this basis, letting $D_t = \{i \in D(x, y) \mid \delta(i) = t\}$ and $x(0) = x$, an induction on $t < m$ proves that $\forall t < m, x(t) \rightarrow x(t+1) = \overline{x(t)}^{D_t}$ is a transition of \mathcal{N} because $D_t \subseteq \mathbf{U}(x(t))$. Thus, $x \rightarrow x(1) \rightarrow \dots \rightarrow x(m-1) \rightarrow y$ is a direct derivation which is asynchronous if \mathbf{H} contains no NOPE-cycles. \square

Informally, Proposition 2 implies the following. When, punctually, m local changes are possible in x , then, unless there is a NOPE-cycle of size m , these m changes can be made asynchronously without risking a deadlock, *i.e.* a situation in which some transitions would have transformed x into a configuration $x(t)$ from which y is not reachable anymore. And, if they can indeed be made asynchronously, then they can be made without resorting to any other changes which are not one of the m intended changes. The next corollary derives from this and sets the backbone of the article. It shows how critical cycles are the main structural aspects of a BAN underlying its possibility to perform synchronous changes that cannot be mimicked asynchronously. First, let us say that $x \rightarrow y$ is *sequentialisable* if it is asynchronous or if it can be broken into a derivation involving smaller transitions $x' \rightarrow y', d(x', y') < d(x, y)$. A synchronous transition $x \rightarrow y$ which is not sequentialisable is called a *normal transition* and it is rather written $x \twoheadrightarrow y$.

Corollary 1. *If $x \twoheadrightarrow y$ is a normal transition, then $D(x, y)$ induces a NOPE-cycle which is x -critical.*

Thus, in a BAN with no NOPE-cycles of size smaller than $m \in \mathbb{N}$, any synchronous change affecting no more than m automata states can be totally sequentialised. We now consider the special case where the only possible critical cycles are Hamiltonian cycles.

Lemma 3. *Let \mathcal{N} be a BAN whose critical cycles all have automata set \mathbf{V} . Then, either \mathcal{N} has a unique normal transition $x \twoheadrightarrow y$, or it has two $x \twoheadrightarrow y$ and $y \twoheadrightarrow x$. In the first case, every $i \in \overline{\mathbf{U}}(y)$ bears a positive loop $(i, i) \in \mathbf{A}$. In both cases, no asynchronous derivations can reach the endpoints of these transitions.*

Proof. Suppose that $x \twoheadrightarrow y$ and $x' \twoheadrightarrow y'$ are two normal transitions. Using Corollary 1, if $y \neq x'$, then $D(x, x') \subsetneq \mathbf{U}(x) = \mathbf{V}$ and $D(x', y) = \mathbf{V} \setminus D(x, x') \subsetneq \mathbf{U}(x') = \mathbf{V}$. In this case, $x \rightarrow x' \rightarrow y$ is a derivation of \mathcal{N} involving smaller transitions than $x \rightarrow y$, in contradiction with $x \rightarrow y$ being normal. Thus, if $x \twoheadrightarrow y$ is not the unique normal transition of \mathcal{N} , then the only other one is $y \twoheadrightarrow x$. For any normal transition $z \twoheadrightarrow z' = \overline{z}^{\mathbf{V}}$, and $\forall i \in \mathbf{V}, z \rightarrow \overline{z'}^i$ is a transition of \mathcal{N} . By hypothesis and by Corollary 1, it is sequentialisable. Since $z \twoheadrightarrow z'$ is not however, this implies that $\forall i \in \mathbf{V}, i \in \overline{\mathbf{U}}(\overline{z'}^i)$. Thus, the endpoint of any normal transition

of \mathcal{N} can be reached by no asynchronous derivations. And eventually, since $\forall i \in \mathbf{V}$, $i \in \overline{\mathbf{U}}(\overline{y}^i)$, any $i \in \overline{\mathbf{U}}(y)$ is such that $\text{sign}(i, i) = +1$ by Lemma 1. \square

4.2 Impact of synchronous transitions

We call *attractors* the terminal SCCs of a transition graph (abusing language because it may be that an attractor doesn't attract anything). Attractors which are not induced by stable configurations (*i.e.* attractors that are induced by several unstable configurations) are called *complex attractors*.

In the sequel, the ATG \mathcal{T}_a is taken as the reference transition graph to which we consider adding a normal synchronous transition. We let $\mathcal{O}_a(x) = \{y \in \mathbb{B}^n \mid x \twoheadrightarrow y\}$ denote the *orbit* of $x \in \mathbb{B}^n$ in \mathcal{T}_a and $\mathcal{B}_a(x) = \{y \in \mathbb{B}^n \mid y \twoheadrightarrow x\}$. $\mathcal{A}_a(x)$ denotes the set of attractors that x can reach in \mathcal{T}_a . We say that a configuration x is *recurrent* when it belongs to an attractor and we denote this attractor by $[x]_a$ (then, $\mathcal{A}_a(x) = \{[x]_a\}$). The *attraction basin* of an attractor $[x]_a$ is $\mathcal{B}_a([x]_a) = \mathcal{B}_a(x) \setminus [x]_a$ [7]. Non-recurrent configurations are called *transient*.

Let $\mathcal{T}_+ = (\mathbb{B}^n, \mathcal{T}_a \cup \{(x, y)\})$ denote the transition graph obtained by adding an arbitrary synchronous transition $x \rightarrow y$ to \mathcal{T}_a . We introduce notations $\mathcal{A}_+(x)$, $\mathcal{B}_+(x)$, $\mathcal{O}_+(x)$ and $[x]_+$ relative to \mathcal{T}_+ similarly to those introduced above for \mathcal{T}_a . We say that an attractor A of \mathcal{T}_a is *destroyed* by $x \rightarrow y$ if all its configurations are transient in \mathcal{T}_+ . On this basis, it is crucial to notice that the addition of $x \rightarrow y$ to \mathcal{T}_a can have several possible consequences on the asymptotic evolutions of \mathcal{N} that go through or start on configuration x .

First of all, it is obvious that if $x \rightarrow y$ is sequentialisable into asynchronous transitions only, adding synchronism through it does not change the asymptotic dynamical behaviour of \mathcal{N} . The case where $x \rightarrow y$ is sequentialisable and is such that the finest sequentialisation maintains one (or more) normal transition(s) is dealt with in what follows. Indeed, from now on and with no loss of generality, let us restrict the study to the case where $x \twoheadrightarrow y$, whose every possible impact induced by its addition on the asymptotic dynamical behaviour of \mathcal{N} is listed below.

1. We say that (the addition of) $x \twoheadrightarrow y$ has *no impact* on the asymptotic evolution of \mathcal{N} when:
 - either x and y are transient in \mathcal{T}_a and $\mathcal{A}_a(y) \subseteq \mathcal{A}_a(x) = \mathcal{A}_+(x)$,
 - or x is transient and y is recurrent in \mathcal{T}_a and $[y]_a \subseteq \mathcal{A}_a(x) = \mathcal{A}_+(x)$,
 - or x and y are both recurrent in \mathcal{T}_a and belong to the same attractor such that $[x]_a = [y]_a$.

Then, the only impact of adding $x \longrightarrow y$ is the possible creation of a shortcut between x and (one or several of) its attractors.

2. We say that transition $x \longrightarrow y$ has *F-impact* (cf. Figure 1) on the asymptotic evolution of \mathcal{N} if x is transient in \mathcal{T}_a and $\mathcal{A}_+(x) = \mathcal{A}_a(x) \cup \mathcal{A}_a(y) \neq \mathcal{A}_a(x)$. Notice that F-impact stands for *Freedom-impact*. Here, $x \longrightarrow y$ causes the growth of the attraction basins of all the attractors of y , consequently adding degrees of freedom to the asymptotic outcomes of the evolutions that pass through x .

Actually, by adding synchronous transitions that have no impact or F-impact, the sets of recurrent configurations of \mathcal{T}_a and \mathcal{T}_+ remain identical.

3. We say that transition $x \longrightarrow y$ has *D-impact* (cf. Figure 2) on the asymptotic evolution of \mathcal{N} if:
 - either x is recurrent and y is transient in \mathcal{T}_a and such that $\mathcal{A}_a(y) \neq \mathcal{A}_a(x) = \{[x]_a\}$,
 - or x and y are both recurrent in \mathcal{T}_a and such that $[x]_a \neq [y]_a$.

Notice that D-impact stands for *Destruction-impact*. In this case, the addition of $x \longrightarrow y$ removes the complex attractor $[x]_a$ by emptying it into the attraction basin of $[y]_a$.

4. We say that transition $x \longrightarrow y$ has *G-impact* (cf. Figure 3) on the asymptotic evolution of \mathcal{N} when x is recurrent and y is transient in \mathcal{T}_a and $\mathcal{A}_a(y) = \mathcal{A}_a(x) = [x]_a$. Notice that G-impact stands for *Growth-impact*. The addition of $x \longrightarrow y$ to \mathcal{T}_a causes attractor $[x]_a$ to absorb all derivations from y to $[x]_a$ and grow into $[x]_+ = [y]_+$ without being destroyed.

It can be checked that the four types of impacts listed above are disjoint and cover all possible cases. Let us emphasise that with the addition of (D-impact) synchronism, a recurrent configuration can become transient. Conversely, the addition of (G-impact) synchronism can turn a transient configuration into a recurrent one. However, synchronism cannot create new attractors from scratch. Furthermore, merging attractors of the ATG requires more than one normal transition.

The addition of $x \longrightarrow y$ to the ATG has no impact or F-impact when x is transient in the ATG (except in the very specific case where x and y belong to the same attractor). To have *significant impact*, i.e. to change the set of attractors of \mathcal{N} (rather than just some of its evolutions towards them), $x \longrightarrow y$ needs to have G- or D-impact. Considering Hamiltonian critical cycles again as in Lemma 3, Lemma 4 below evidences that having this sort of impact requires to embed critical cycles in a larger, structural environment.

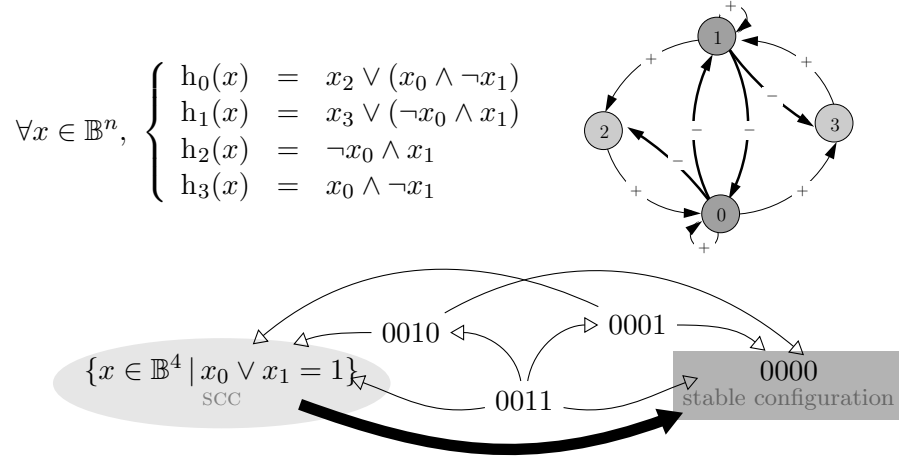


Figure 2: Top left: mechanisms of BAN $\mathcal{N}^* = \{h_i \mid i \in \mathbf{V} = \{0, \dots, 3\}\}$. Top right: signed structure of \mathcal{N}^* . Bottom: reduced version of \mathcal{T}_+ , which is the transition graph obtained by adding the normal transition $1100 \rightarrow 0000$ to the ATG \mathcal{T}_a of \mathcal{N}^* . The shaded ellipse corresponds to a SCC which is terminal in \mathcal{T}_a but not in \mathcal{T}_+ nor in the ETG \mathcal{T} . This illustrates that $1100 \rightarrow 0000$ has D-impact on the asymptotic evolution of \mathcal{N}^* . It follows from Corollary 1 of Section 4.1 that this is essentially due to the positive cycle of length 2 induced by $\{0, 1\} \subseteq \mathbf{V}$.

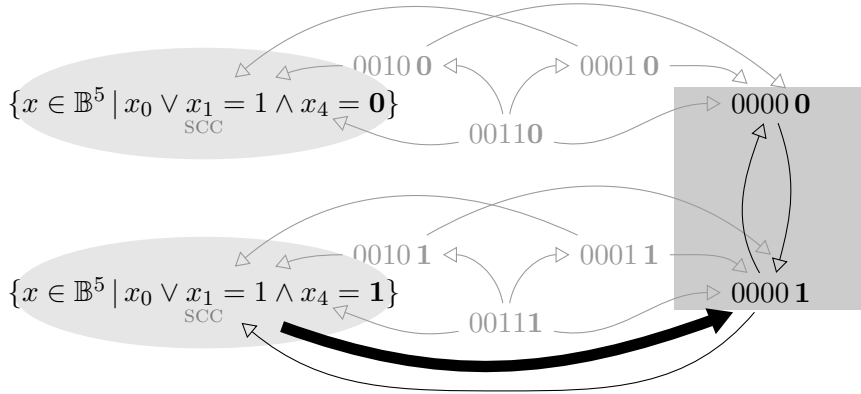


Figure 3: Reduced version of \mathcal{T}_+ obtained by adding normal transition $11001 \rightarrow 00001$ to the ATG of \mathcal{N} . \mathcal{N} is the BAN that is obtained from \mathcal{N}^* (cf. Figure 2) as follows. A fifth automaton $i = 4 \in \mathbf{V}$ is added such that $f_4(x) = \neg(x_0 \vee x_1 \vee x_2 \vee x_3) \wedge \neg x_4$, and we let $f_0(x) = h_0(x) \vee (x_4 \wedge \neg x_1 \wedge \neg x_3)$ and $\forall i \in \{1, 2, 3\}, f_i(x) = h_i(x)$. It can be checked that $h_0(x) \neq f_0(x) \Rightarrow x = 00001$. $11001 \rightarrow 00001$ therefore has G-impact.

Lemma 4. *Let \mathcal{N} be a BAN with no normal transitions of size smaller than its size n . Then, any transition $x \blacktriangleright y$ either has no impact on the asymptote of \mathcal{N} or it has F-impact. In the latter case, y is stable and has an empty basin $\mathcal{B}_a(y) = \emptyset$ in the ATG, and all automata of \mathcal{N} have a positive loop.*

Proof. Let $x \blacktriangleright y = \bar{x}^{\mathbf{V}}$. Notice that for all $i \in \mathbf{V}$, there inevitably exists a synchronous transition $x \rightarrow \bar{y}^i$, because $D(x, \bar{y}^i) \subseteq \mathbf{U}(x) = \mathbf{V}$. By hypothesis, since there are no normal transitions of size smaller than n , this synchronous transition is sequentialisable into asynchronous transitions and we get that $\forall i \in \mathbf{V}$, $x \rightarrow \bar{y}^i$. Thus, $\forall z \in \mathbb{B}^n$, $y \rightarrow z \Rightarrow x \rightarrow z$. As a consequence:

- either $\mathbf{U}(y) \neq \emptyset$ in which case $\mathcal{A}_a(y) \subseteq \mathcal{A}_a(x)$. In this case, $x \blacktriangleright y$ has no impact on the asymptotic behaviour of \mathcal{N} ;
- or $\mathbf{U}(y) = \emptyset$ and thus, y is stable. In this case, $x \blacktriangleright y$ has F-impact. Moreover, obviously $y \blacktriangleright x$ does not exist. Consequently, Lemma 3 holds and $\mathcal{B}_a(y) = \emptyset$ and $\forall i \in \mathbf{V}$, $\text{sign}(i, i) = +1$.

□

4.3 Synchronism-sensitivity

Let us say that \mathcal{N} has *no synchronism-sensitivity* if none of its normal transitions has any impact (*cf.* Section 4.2), that it has *little sensitivity* if it admits normal transitions with F-impact, and that it has *significant sensitivity* if it admits normal transitions with significant impact, *i.e.* D- or G-impact.

Theorem 1. *1) Synchronism-sensitivity requires the existence of a critical cycle, and thus of a NOPE-cycle.*

2) Significant sensitivity requires the existence of a critical cycle of length strictly smaller than the BAN size as well as of a negative cycle.

3) In the absence of a Hamiltonian NOPE-cycle and positive loops on all automata, little sensitivity also requires a critical cycle of length strictly smaller than the BAN size.

Proof. 1) Synchronism-sensitivity requires a normal transition which requires a critical NOPE-cycle by Corollary 1.

2) A normal transition $x \blacktriangleright y$ with significant impact (*cf.* Section 4.2) must be induced by a non-Hamiltonian critical cycle (*cf.* Lemma 4). And since x must be recurrent, there must be a negative cycle to induce the complex attractor $[x]_a$ [23].

3) By Lemma 4.

□

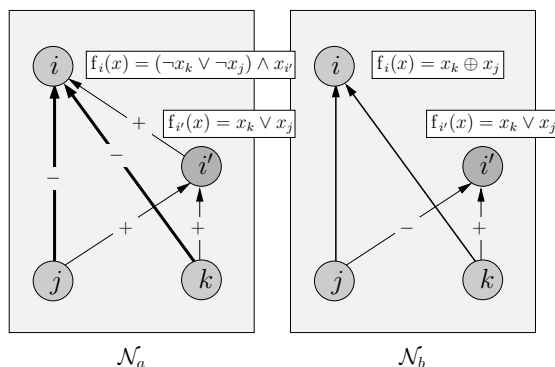


Figure 4: In the monotonic BAN \mathcal{N}_a , if the state change of i' occurs before that of i in all $x \in \mathbb{B}^4$, then, i takes state $f_0(x_j, x_k, f_{i'}(x)) = x_j \oplus x_k$. Imposing systematically the immediate precedence of i' state changes over i state changes makes the mechanism $f_i \in \mathcal{N}_a$ of i non-monotonic. As a result, \mathcal{N}_a behaves like the non-monotonic BAN \mathcal{N}_b .

5 Conclusion and perspectives

We have evidenced that synchronism in itself may impact significantly on the asymptotic behaviour of a network. It cannot only modify transient behaviours and make attractors grow but also destroy complex attractors. And it can have this effect by filtering out local instabilities that asynchronism cannot get rid of, and, as a result, decrease the global system instability. This contradicts a misconception existing in some theoretical modelling fields (that may have contributed also to the little attention that has been paid to synchronism in these fields) assuming that asynchronism guarantees a greater global stability while synchronism's main impact consists in artificially entertaining instabilities.

Exploiting the interpretation of synchronism discussed in Section 2.3, the existence of BANs that are significantly sensitive to synchronism suggests that time may indeed be a determining parameter in itself. The particular conditions that are necessary for a BAN to be sensitive to synchronism, however (*cf.* Theorem 1), suggest that most systems are not affected by time in the following sense. Punctually, their possible future evolutions do not depend on the temporal parameters (such as the durations and relative speeds of possible changes) that are responsible of the organisation of possible changes.

Interestingly, it can be checked that the smallest *monotonic* BANs that have significant synchronism-sensitivity have size 3 and are monotonic encodings of the sensitive BANs of size 2 which all are non-monotonic. Moreover, a closer examination of the h_is in Figure 2 shows that the critical synchronous transition $x = 1100 \xrightarrow{\text{h}_i} \bar{x}^{\{0,1\}} = 0000$ of the corresponding network

“dynamically” mimics a \vee connector that combines $h_0(x)$ and $h_1(x)$ so as to produce an effect (witnessed by automata 2 and 3) similar to the most basic non-monotonic mechanism: $x \mapsto x_0 \oplus x_1 = (x_0 \wedge \neg x_1) \vee (\neg x_0 \wedge x_1)$. Informally, this, and Figure 4, suggest a more general notion of non-monotonicity that takes into account how state changes are organised relatively in time (and by which BAN \mathcal{N}_a of Figure 4 would be considered non-monotonic). Non-monotonicity would then amount to the existence of paths, in the structure of a BAN, of different signs and identical extremities. We conjecture that significantly sensitive BANs that admit D-impact are non-monotonic in this larger sense, which can be formalised as below by basing ourselves on a notion of intrinsic simulation stating that a BAN \mathcal{N}_1 simulates a BAN \mathcal{N}_2 if and only if, (i) the mechanisms of \mathcal{N}_1 are defined as functions of those of \mathcal{N}_2 , and (ii) if \mathcal{N}_2 has k stable configurations and ℓ complex attractors, then \mathcal{N}_1 also admits k stable configurations and ℓ complex attractors (in other terms, there exists a bijection between the set of attractors of \mathcal{N}_1 into that of attractors of \mathcal{N}_2).

Conjecture *If $\mathcal{N} = \{f_i : \mathbb{B}^n \rightarrow \mathbb{B} \mid i \in \{0, \dots, n-1\}\}$ is a monotonic BAN that admits a normal transition $x \xrightarrow{\text{D}} y$ having D-impact, then there exists a non-monotonic BAN $\mathcal{N}' = \{g_i : \mathbb{B}^m \rightarrow \mathbb{B} \mid i \in \{0, \dots, m-1\}$ and $m < n\}$, such that at least one g_i is defined as the composition of transition rules f_j s and such that \mathcal{N}' simulates \mathcal{N} .*

In other terms, based on the work presented in this article, our intuition is that in the absence of contradicting signals circulating between the two same entities (one of which being the emitter of both signals, the other being the receptor), the implementation of mechanisms can be stretched out or concentrated in time with “little impact” (*i.e.* with no impact at all or F-impact) on the final asymptotic behaviour of the network.

To end on another closely related perspective of this work, let us note that Proposition 2 highlights an interesting fact: when a certain number of changes are possible all at once, if the whole set of these changes can be sequentialisable into asynchronous transitions (which happens in most cases as we have proven), then it can be made without ever needing any reversed changes, namely, changes whose sole purpose is to unlock the intended changes (cf. the formal definition page 9). We actually lack intuitions that could support the rarity of computational mechanisms relying on reversed changes. For this reason, we believe that an important sequel to the work presented here is to determine (structural) conditions that are necessary for the existence of indirect derivations from one configuration to another when there are no direct ones, *i.e.* the conditions that explain “a need for reversed changes”.

Acknowledgements This work has been partially supported by the project PACA APEX FRI.

References

- [1] Aldana, M.: Boolean dynamics of networks with scale-free topology. *Physica D* **185**, 45–66 (2003)
- [2] Aracena, J., Demongeot, J., Goles, E.: On limit cycles of monotone functions with symmetric connection graph. *Theoretical Computer Science* **322**, 237–244 (2004)
- [3] Aracena, J., Demongeot, J., Goles, E.: Positive and negative circuits in discrete neural networks. *IEEE Transactions on Neural Networks* **15**, 77–83 (2004)
- [4] Chandesris, J., Dennunzio, A., Formenti, E., Manzoni, L.: Computational Aspects of Asynchronous Cellular Automata. In: Proceedings of DLT, *Lecture Notes in Computer Science*, vol. 6795, pp. 466–468. Springer (2011)
- [5] Choffrut, C. (ed.): Automata networks, *Lecture Notes in Computer Science*, vol. 316. Springer-Verlag (1988)
- [6] Combe, P., Nencka, H.: Frustration and overblocking on graphs. *Mathematical and Computer Modelling* **26**, 307–309 (1997)
- [7] Cosnard, M., Demongeot, J.: Iteration Theory and its Functional Equations, *Lecture Notes in Mathematics*, vol. 1163, chap. On the definition of attractors, pp. 23–31. Springer (1985)
- [8] Demongeot, J., Goles, E., Morvan, M., Noual, M., Sené, S.: Attraction basins as gauges of robustness against boundary conditions in biological complex systems. *PLoS One* **5**, e11,793 (2010)
- [9] Derrida, B., Pomeau, Y.: Random networks of automata: a simple annealed approximation. *Europhysics Letters* **1**, 45–49 (1986)
- [10] Fatès, N.: Experimental study of elementary cellular automata dynamics using the density parameter. *Discrete Mathematics and Theoretical Computer Science* **AB**, 155–166 (2003)
- [11] Floréen, P.: Computational complexity problems in neural associative memories. Ph.D. thesis, University of Helsinki (1992)
- [12] Goles, E., Martínez, S.: Neural and automata networks: dynamical behaviour and applications. Kluwer Academic Publishers (1990)
- [13] Goles, E., Olivos, J.: Comportement périodique des fonctions à seuil binaires et applications. *Discrete Applied Mathematics* **3**, 93–105 (1981)

- [14] Goles, E., Salinas, L.: Comparison between parallel and serial dynamics of Boolean networks. *Theoretical Computer Science* **396**, 247–253 (2008)
- [15] Harvey, I., Bossomaier, T.: Time out of joint: attractors in asynchronous random Boolean networks. In: *Proceedings of ECAL*, pp. 67–75. MIT Press (1997)
- [16] Kauffman, S.A.: Metabolic stability and epigenesis in randomly constructed genetic nets. *Journal of Theoretical Biology* **22**, 437–467 (1969)
- [17] Kleene, S.C.: Automata Studies, *Annals of Mathematics Studies*, vol. 34, chap. Representation of events in nerve nets and finite automata, pp. 3–41. Princeton University Press (1956)
- [18] Manzoni, L.: Asynchronous cellular automata and dynamical properties. *Natural Computing* **11**, 269–276 (2012)
- [19] McCulloch, W.S., Pitts, W.H.: A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics* **5**, 115–133 (1943)
- [20] Noual, M., Regnault, D., Sené, S.: Boolean networks synchronism sensitivity and XOR circulant networks convergence time. In: *Full Papers Proceedings of AUTOMATA & JAC'2012, Electronic Proceedings in Theoretical Computer Science*, vol. 90, pp. 37–52. Open Publishing Association (2012)
- [21] Noual, M., Regnault, D., Sené, S.: About non-monotony in Boolean automata networks. *Theoretical Computer Science* **504**, 12–25 (2013)
- [22] Remy, É., Ruet, P.: From minimal signed circuits to the dynamics of Boolean regulatory networks. *Bioinformatics* **24**, i220–i226 (2008)
- [23] Richard, A.: Negative circuits and sustained oscillations in asynchronous automata networks. *Advances in Applied Mathematics* **44**, 378–392 (2010)
- [24] Robert, F.: *Discrete iterations: a metric study*. Springer Verlag (1986)
- [25] Saint Savage, N.: The effects of state dependent and state independent probabilistic updating on Boolean network dynamics. Ph.D. thesis, University of Manchester (2005)
- [26] Schabanel, N., Regnault, D., Thierry, É.: Progresses in the analysis of stochastic 2D cellular automata: a study of asynchronous 2D minority. *Theoretical Computer Science* **410**, 4844–4855 (2009)

- [27] Thieffry, D., Thomas, R.: Dynamical behaviour of biological regulatory networks—II. Immunity control in bacteriophage lambda. *Bulletin of Mathematical Biology* **57**, 277–297 (1995)
- [28] Thomas, R.: Boolean formalization of genetic control circuits. *Journal of Theoretical Biology* **42**, 563–585 (1973)
- [29] Thomas, R.: Regulatory networks seen as asynchronous automata: a logical description. *Journal of Theoretical Biology* **153**, 1–23 (1991)
- [30] Toulouse, G.: Theory of the frustration effect in spin glasses. I. *Communications on Physics* **2**, 115–119 (1977)
- [31] Vannimenus, J., Toulouse, G.: Theory of the frustration effect in spin glasses. II. Ising spins on a square lattice. *Journal of Physics C* **10**, L537–L542 (1977)
- [32] von Neumann, J.: *Theory of self-reproducing automata*. University of Illinois Press (1966)