Advances in Applied Mathematics 45 (2010) 346–358

Contents lists available at ScienceDirect

Advances in Applied Mathematics

www.elsevier.com/locate/yaama

Sequential operator for filtering cycles in Boolean networks

Eric Goles [b,c,1], Lilian Salinas [a,*,1]

[a] Department of Computer Sciences, University of Concepción, Edmundo Larenas 215, Piso 3, Concepción, Chile
[b] Universidad Adolfo Ibañez, Av. Diagonal Las Torres 2640, Peñalolen, Santiago, Chile
[c] Institute for Complex Systems (ISCV), Av. Artillería 600B, Cerro Artillería, Valparaíso, Chile

ARTICLE INFO

Article history:
Received 10 May 2009
Accepted 29 January 2010
Available online 31 March 2010

MSC:
68Q25
68Q17
68Q80
37N25
92B05

Keywords:
Boolean network
Attractor
Fixed point
Dynamical cycle
Filter

ABSTRACT

Given a Boolean network without negative circuits, we propose a polynomial algorithm to build another network such that, when updated in parallel, it has the same fixed points than the original one, but it does not have any dynamical cycle. To achieve that, we apply a network transformation related to the sequential update. As a corollary, we can find a fixed point in polynomial time for this kind of networks.

1. Introduction

Boolean networks were introduced by Kauffman [8] to model genetic regulatory networks. Boolean networks are applied in many areas including circuit theory, computer science [5,16] and molecular biology [7,8]. These networks are defined by a set of states, a transition function and an update schedule. In biology, fixed points and dynamical cycles (attractors) can represent a memory trace, a pattern of motor nerve activity, a state of an immune network or a cell type. For example, in the

* Corresponding author.
E-mail addresses: eric.chacc@uai.cl (E. Goles), lilisalinas@udec.cl (L. Salinas).

[1] This work was partially supported by CONICYT FONDAP program in applied mathematics, FONDECYT project 1100003, Basal-CMM, STIC-AMSUD project and MORPHEX (E. Goles).

0196-8858/$ – see front matter
doi:10.1016/j.aam.2010.03.002

modeling of gene regulatory networks, the attractors are associated to different cell states defined by patterns of gene activity.

One of the first theoretical studies to compare different update schedules was done by F. Robert [12]. He proves that in Boolean networks without circuits there exists a unique fixed point. He also applied the Gauss-Seidelization process on Boolean networks without circuits, proving that this process becomes stationary and that the resulting Boolean network has the same fixed points than the original one ([12], Chapter 4, Theorem 7). The Gauss-Seidelization is equivalent to the successive application of a certain sequential operator on a Boolean network.

Here we focus our attention on the successive application of this sequential operator on Boolean networks where all the circuits are positive. We prove that the Boolean network resulting from this process has only fixed points as attractors; i.e., this process "filters out" the cycles; furthermore, the fixed points are reached in at most n updates. It is important to note that we can find only one fixed point in polynomial time, but not all of them. The problem of finding all the fixed points of a Boolean network is NP-hard even for monotonic Boolean networks [17].

We use a class of Boolean networks, called regulatory Boolean networks [2], where each interaction between the elements of the network corresponds either to a positive or to a negative interaction. This family of networks includes the Boolean networks with hierarchically canalizing functions, also known as nested canalizing functions (Kauffman et al. [9]), introduced by Szallasi and Liang [14] and recently studied by Nikolajewa et al. [11]. A great amount of data collected by Harris et al. [6] about the updating rules of different real genes, and some models of real genetic networks (for example, Aracena [3]; Aracena et al. [4]; Mendoza and Alvarez-Buylla [10]; Sánchez and Thieffry [13]) show that interactions between gene and gene products are mainly of the activation or inhibition type.

Section 2 presents the definitions and notations used in the article. In Section 3, we prove some preliminary results. These results are very useful in the proof of the theorem in Section 4, which is the main result of this article. This theorem proves the existence of a polynomial algorithm to "filter out" the dynamical cycles in a Boolean network with a graph without negative circuits. Finally, in Section 5, we discuss the scope of the application of this algorithm on Boolean networks.

2. Notation and definitions

A Boolean network N_F is defined by a finite set of variable states $\{x_1, \ldots, x_n\}$, where $x_i \in \{0, 1\}$ and a global transition function $F : \{0, 1\}^n \to \{0, 1\}^n$, where $F(x) = (f_1(x), \ldots, f_n(x))$ and $x = (x_1, \ldots, x_n)$, $x_i \in \{0, 1\}$.

By default a Boolean network is updated in a parallel way, i.e.:

$$x^{r+1} = F\left(x^r\right) \tag{1}$$

where $x^0 \in \{0, 1\}^n$.

However, alternative update schedules are also used. A common one is the sequential update, defined by:

$$\begin{aligned} x_1^{r+1} &= f_1\left(x_1^r, \ldots, x_n^r\right), \\ x_i^{r+1} &= f_i\left(x_1^{r+1}, \ldots, x_{i-1}^{r+1}, x_i^r, \ldots, x_n^r\right), \quad \forall i = 2, \ldots, n. \end{aligned} \tag{2}$$

The sequential update is equivalent to applying a function $F^{(1)} : \{0, 1\}^n \to \{0, 1\}^n$ in a parallel way, where $F^{(1)}(x) = (f_1^{(1)}(x), \ldots, f_n^{(1)}(x))$ is defined by:

$$\begin{aligned} f_1^{(1)}(x) &= f_1(x), \\ f_i^{(1)}(x) &= f_i\left(f_1^{(1)}(x), \ldots, f_{i-1}^{(1)}(x), x_i, \ldots, x_n\right), \quad \forall i = 2, \ldots, n. \end{aligned} \tag{3}$$

We define F^u as the function which describes the dynamical behavior of the network after taking the update schedule into account. Thus, $F^u = F$ when the update is parallel and $F^u = F^{(1)}$ when it is sequential.

Since $\{0,1\}^n$ is a finite set we have two possible limit behaviors for the iteration of a network:

- *Fixed point*: a vector $x \in \{0,1\}^n$ such that $F^u(x) = x$.
- *Cycle*: a sequence $[x^0, \dots, x^{p-1}, x^0]$, $p > 1$, such that $x^j \in \{0,1\}^n$, x^j are pairwise distinct and $F^u(x^j) = x^{j+1}$, for all $j = 0, \dots, p-2$ and $F^u(x^{p-1}) = x^0$.

Fixed points and cycles are called *attractors* of the network.

The *graph associated* to N_F is the directed digraph $G^F = (V, A)$, where:

- $V = \{1, \dots, n\}$.
- $(i, j) \in A$ if and only if f_j depends on x_i, i.e., if there exists $x \in \{0,1\}^n$ such that

$$f_j(x_1, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n) \neq f_j(x_1, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n).$$

The node set of G^F is referred to as $V(G^F)$, its arc set as $A(G^F)$.

A *circuit* of G^F is a sequence of different nodes (except the extreme ones) $i_1, i_2, \dots, i_k, i_1$ of V where $(i_l, i_{l+1}) \in A$ for $l = 1, \dots, k-1$ and $(i_k, i_1) \in A$. k is the length of the circuit. A *loop* is a circuit of length one.

A Boolean function $f : \{0,1\}^n \to \{0,1\}$ is increasing monotonic on input i if

$$\forall x \in \{0,1\}^n, \quad f(x_1, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n) \leqslant f(x_1, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n),$$

and decreasing monotonic on input i if

$$\forall x \in \{0,1\}^n, \quad (x_1, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n) \geqslant f(x_1, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n).$$

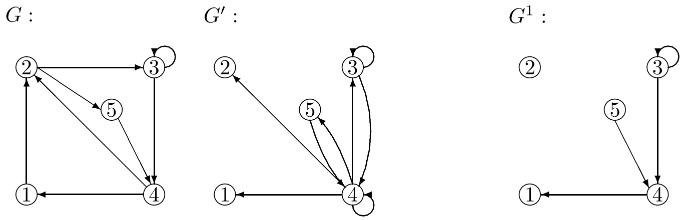
A Boolean function $f : \{0,1\}^n \to \{0,1\}$ is said to be a sign-definite function, also known as unate function (see [1]), if for each $i = 1, \dots, n$, it is either increasing monotonic or decreasing monotonic on input i. Equivalently, a Boolean function is sign-definite if it can be represented by a formula in disjunctive normal form in which all occurrences of any given literal are either negated or nonnegated [1]. Examples of sign-definite Boolean functions are the hierarchically canalizing functions [9,11], where the literals appear once in the disjunctive normal form. Another example of sign-definite functions are the threshold Boolean functions, where the sign of the weight of each interaction is associated to the type of monotony. On the other hand, there are sign-definite functions which are neither hierarchically canalizing functions nor threshold functions, for example: $f(x_1, x_2, x_3, x_4) = (x_1 \wedge x_2) \vee (x_3 \wedge x_4)$. A well-known example of non-sign-definite Boolean function is "$\veebar$" (XOR), that is, $x_1 \veebar x_2 = (\neg x_1 \wedge x_2) \vee (x_1 \wedge \neg x_2)$. An analysis of Harris' collected data revealed that 134 of the 139 rules are sign-definite functions.

Let $F : \{0,1\}^n \to \{0,1\}^n$ be defined by $F(x) = (f_1(x), \dots, f_n(x))$, where $f_i : \{0,1\}^n \to \{0,1\}$, for all $i = 1, \dots, n$. We say that F is a sign-definite function, if for all $i = 1, \dots, n$, f_i is a sign-definite function.

Given a sign-definite function $F : \{0,1\}^n \to \{0,1\}^n$, such that $F(x) = (f_1(x), \dots, f_n(x))$, we denote by $I^+(j)$ and $I^-(j)$ the set of indices where f_j is increasing monotonic and decreasing monotonic respectively.

Hence, for every Boolean network N where F is a sign-definite function, we can define a sign function $w_F : A(G^F) \to \{-1, 1\}$ with

$$w_F(i, j) = \begin{cases} -1 & \text{if } i \in I^-(j), \\ +1 & \text{if } i \in I^+(j). \end{cases}$$



N :
$f_1(\vec{x}) = x_4$
$f_2(\vec{x}) = x_1 \vee \neg x_4$
$f_3(\vec{x}) = x_2 \wedge x_3$
$f_4(\vec{x}) = x_3 \wedge x_5$
$f_5(\vec{x}) = \neg x_2$

$f_1^{(1)}(\vec{x}) = x_4$
$f_2^{(1)}(\vec{x}) = x_4 \vee \neg x_4$
$f_3^{(1)}(\vec{x}) = (x_4 \vee \neg x_4) \wedge x_3$
$f_4^{(1)}(\vec{x}) = x_3 \wedge (x_4 \vee \neg x_4) \wedge x_5$
$f_5^{(1)}(\vec{x}) = \neg(x_4 \vee \neg x_4)$

$N^1 = \mathcal{S}(N)$
$f_1^{(1)}(\vec{x}) = x_4$
$f_2^{(1)}(\vec{x}) = 1$
$f_3^{(1)}(\vec{x}) = x_3$
$f_4^{(1)}(\vec{x}) = x_3 \wedge x_5$
$f_5^{(1)}(\vec{x}) = 0$

Fig. 1. Graph associated to $\mathcal{S}(N)$. Between G' and G^1 only the effective dependences are kept.

An arc $(i, j) \in A(G^F)$ will be called positive if $w_F(i, j) = 1$ and negative otherwise. We will say that a path is positive if the number of its negative arcs is even, and negative otherwise. (G^F, w_F) will be called graph with sign of N.

We denote by $\mathcal{N}_n = \{N_F / F : \{0,1\}^n \to \{0,1\}^n\}$ the set of Boolean networks with n vertices in their associated graphs. The operator $\mathcal{S} : \mathcal{N}_n \to \mathcal{N}_n$ will be defined as $\mathcal{S}(N_F) = N_{F^{(1)}}$, where $F^{(1)}(x) = (f_1^{(1)}(x), \ldots, f_n^{(1)}(x))$ is given by (3); hence, this operator simulates in parallel the behavior of the network N_F *as if* it was updated in a sequential way. It corresponds to the Gauss–Seidel operator defined in [12].

We define recursively the k-sequential network of N_F denoted by $N^k = \mathcal{S}(N^{k-1})$, where $N^0 = N_F$. Accordingly, we denote $F = F^{(0)}$.

We see that we can define the graph associated to the sequential Boolean network, i.e., if $G = (V, A)$ is the graph associated to N we will call $G^1 = (V, A^1)$ the graph associated to $N^1 = \mathcal{S}(N)$. We observe that the set of vertex is the same in both cases, but the set of arcs is different. In fact we can see that:

$$(i, j) \in A^1 \implies \big(i \geqslant j \wedge (i, j) \in A\big) \vee \big((\exists k < j), (i, k) \in A^1 \wedge (k, j) \in A\big). \tag{4}$$

Fig. 1 shows on the left a Boolean network N and its associated graph, on the right the effective graph associated to $\mathcal{S}(N)$, and on the center the graph described by the right side of (4) (i.e., before simplification).

3. Boolean networks without negative circuits

In this section we will restrict our attention to the Boolean networks with associated graph without negative circuits, and we prove some lemmas required for the proof of Theorem 5.

Lemma 1. *Let N be a Boolean network such that all the circuits are positive, and let $N^1 = \mathcal{S}(N)$ be the sequential Boolean network of N. Then all the circuits of N^1 are positive.*

Proof. Let $G = (V, A)$ be the graph associated to N and $G^1 = (V, A^1)$ the graph associated to N^1.

To prove this lemma we need to prove that if $(i, j) \in A^1$ then there exists a path $i_0 i_1 \ldots i_k$ in G where $i_0 = i$ and $i_k = j$, and if all the paths from i to j have the same sign then $w_{F^{(1)}}(i, j) = \prod_{l=1}^{k} w_F(i_{l-1}, i_l)$. In this way, we prove that any path in G^1, in particular the circuits, comes from a path in G, and the sign of the path in G is related to the sign of the path in G.

We know that:

$$f_j^{(1)}(x) = f_j\big(f_1^{(1)}, \ldots, f_{j-1}^{(1)}, x_j, \ldots, x_n\big).$$

Then:

$$(i, j) \in A^1 \quad \Longrightarrow \quad \big(i \geqslant j \wedge (i, j) \in A\big) \vee \big((\exists k < j),\ (i,k) \in A^1 \wedge (k, j) \in A\big).$$

By induction on j, we prove that for all $(i, j) \in A^1$ there exists a path from i to j in G.

Basis. For $j = 1$, if $(i, 1) \in A^1$, then $(i, 1) \in A$, thus there exists a path of length one between i and j in G. The sign of this path is the sign of the arc $(i, 1) \in A$.

Induction hypothesis. We suppose that for all $k \leqslant j$, if $(i,k) \in A^1$, then there exists a path from i to k in G. And, if all the paths from i to k have the same sign in G then the sign of (i,k) in G^1 is the sign of the paths.

Case *j* + 1. Given $(i, j + 1) \in A^1$, we prove that there exists a path from i to $j + 1$ in G. We study two cases:

1. If $i \geqslant j + 1$ and $(i, j + 1) \in A$ we have a path of length one in G.
2. if $(i, j + 1)$ is such that there exists $k < j + 1, (i,k) \in A^1, (k, j + 1) \in A$, then by induction hypothesis there exists a path from i to k in G. Since $(k, j + 1) \in A$, there exists a path from i to $j + 1$.

Now, we prove that if all the paths from i to j have the same sign in G, then the sign of (i, j) in G^1 is the sign of the paths. We give the proof for the case where the sign of the paths is positive, the negative case is analogous. We prove that if the sign of path is positive then the function $f_{j+1}^{(1)}$ is increasing on input i:

$$f_{j+1}^{(1)}(x) = f_{j+1}\big(f_1^{(1)}(x), \ldots, f_j^{(1)}(x), x_{j+1}, \ldots, x_n\big).$$

1. If $i \geqslant j + 1$ and $(i, j + 1) \in A$ we have a path of length one in G and $w_F(i, j + 1) = +1$ then f_{j+1} is increasing on input i, and thus $f_{j+1}^{(1)}$ is increasing on input i.
2. if $(i, j + 1)$ is such that there exists $k < j + 1, (i,k) \in A^1, (k, j + 1) \in A$; then by induction hypothesis there exists a path from i to k in G and $w_{F^{(1)}}(i,k) = \prod_{l=1}^{p} w_F(i_{l-1}, i_l)$. Since the path is positive if $w_{F^{(1)}}(i,k) = +1$ then $w_F(k, j + 1) = +1$. Therefore $f_k^{(1)}$ is increasing on input i and f_{j+1} is increasing on input k, and hence $f_{j+1}^{(1)}$ is increasing on input i. Now, if $w_{F^{(1)}}(i,k) = -1$ then $w_F(k, j + 1) = -1$. Therefore $f_k^{(1)}$ is decreasing on input i and f_{j+1} is decreasing on input k, and then $f_{j+1}^{(1)}$ is increasing on input i.

As a corollary of the last lemma, we can conclude that in a Boolean network with associated graph without negative circuits all the loops in the k-sequential network are positive. This point is very important, because the fact that the loops are positive is one of the hypothesis of the next lemma, which is the base of the proof of the main result of this article.

Lemma 2. *Let N be a Boolean network such that its parallel and sequential dynamics are identical and all the loops are positive. Then, all the attractors of the network are fixed points, and they are reached in at most n updates. In fact, if we denote* $\underbrace{F \circ \cdots \circ F}_{k\ \text{times}}(x^0) = x^k$, *the following holds*:

$$x_{n-l}^{l+1} = x_{n-l}^k, \quad \forall k > l,\ l = 0, \ldots, n - 1.$$

Proof. We proceed by induction on l.

Basis. In first place we prove that $x_n^k = x_n^{k+1}$ implies $x_n^p = x_n^k$, $\forall p \geqslant k$. Since the sequential dynamics is equal to the parallel dynamics:

$$x_n^{k+1} = f_n\big(x_1^{k+1}, \dots, x_{n-1}^{k+1}, x_n^k\big) \qquad \text{Sequential update},$$
$$x_n^{k+2} = f_n\big(x_1^{k+1}, \dots, x_{n-1}^{k+1}, x_n^{k+1}\big) \qquad \text{Parallel update}.$$

Hence if $x_n^k = x_n^{k+1}$ then $x_n^{k+1} = x_n^{k+2}$ and in this way it is clear that, $x_n^p = x_n^k$, $\forall p \geqslant k$.

Now we show, $x_n^1 = x_n^2$. If $x_n^0 = x_n^1$, the previous sentence applies. Let us suppose $x_n^0 = x$, $x_n^1 = \neg x$, then

$$x_n^1 = f_n\big(x_1^1, \dots, x_{n-1}^1, x\big) = \neg x \qquad \text{Sequential update},$$
$$x_n^2 = f_n\big(x_1^1, \dots, x_{n-1}^1, \neg x\big) \qquad \text{Parallel update}.$$

If f_n does not depend on x_n then $x_n^2 = x_n^1$. If f_n depends on x_n, f_n is increasing monotonic respect to x_n, then $x_n^2 = \neg x = x_n^1$.

Induction hypothesis.

$$x_{n-j}^{j+1} = x_{n-j}^k, \quad \forall k > j,\ j = 0, \dots, l-1.$$

Case l. In first place we prove that $x_{n-l}^{l+1} = x_{n-l}^{l+2}$ implies $x_{n-l}^p = x_{n-l}^{l+1}$, $\forall p > l$. Since the sequential update is equal to the parallel update:

$$x_{n-l}^{l+2} = f_{n-l}\big(x_1^{l+2}, \dots, x_{n-l-1}^{l+2}, x_{n-l}^{l+1}, \dots, x_n^{l+1}\big) \qquad \text{Sequential update},$$
$$x_{n-l}^{l+3} = f_{n-l}\big(x_1^{l+2}, \dots, x_{n-l-1}^{l+2}, x_{n-l}^{l+2}, \dots, x_n^{l+2}\big) \qquad \text{Parallel update}.$$

By induction hypothesis $x_{n-j}^{l+1} = x_{n-j}^{l+2}$ $\forall j < l$, then:

$$x_{n-l}^{l+2} = f_{n-l}\big(x_1^{l+2}, \dots, x_{n-l-1}^{l+2}, x_{n-l}^{l+1}, x_{n-l+1}^{l+1}, \dots, x_n^{l+1}\big),$$
$$x_{n-l}^{l+3} = f_{n-l}\big(x_1^{l+2}, \dots, x_{n-l-1}^{l+2}, x_{n-l}^{l+2}, x_{n-l+1}^{l+1}, \dots, x_n^{l+1}\big).$$

Then if $x_{n-l}^{l+1} = x_{n-l}^{l+2}$, this implies $x_{n-l}^p = x_{n-l}^{l+1}$, $\forall p > l$.

Now, we prove that $x_{n-l}^{l+1} = x_{n-l}^{l+2}$:

$$x_{n-l}^{l+1} = f_{n-l}\big(x_1^{l+1}, \dots, x_{n-l-1}^{l+1}, x_{n-l}^{l}, x_{n-l+1}^{l}, \dots, x_n^{l}\big) \qquad \text{Sequential update},$$
$$x_{n-l}^{l+2} = f_{n-l}\big(x_1^{l+1}, \dots, x_{n-l-1}^{l+1}, x_{n-l}^{l+1}, x_{n-l+1}^{l+1}, \dots, x_n^{l+1}\big) \qquad \text{Parallel update}.$$

By induction hypothesis:

$$x_{n-l}^{l+1} = f_{n-l}\big(x_1^{l+1}, \dots, x_{n-l-1}^{l+1}, x_{n-l}^{l}, x_{n-l+1}^{l}, \dots, x_n^{l}\big),$$
$$x_{n-l}^{l+2} = f_{n-l}\big(x_1^{l+1}, \dots, x_{n-l-1}^{l+1}, x_{n-l}^{l+1}, x_{n-l+1}^{l}, \dots, x_n^{l}\big).$$

We have three possibilities:

1. If f_{n-l} does not depend on x_{n-l} then $x_{n-l}^{l+1} = x_{n-l}^{l+2}$.
2. If f_{n-l} depends on x_{n-l} and $x_{n-l}^{l} = x_{n-l}^{l+1}$ then $x_{n-l}^{l+1} = x_{n-l}^{l+2}$, by sentence showed above.
3. If f_{n-l} depends on x_{n-l} and $x_{n-l}^{l} = x$, $x_{n-l}^{l+1} = \neg x$

$$\neg x = f_{n-l}\bigl(x_1^{l+1}, \ldots, x_{n-l-1}^{l+1}, x, x_{n-l+1}^{l}, \ldots, x_n^{l}\bigr),$$
$$x_{n-l}^{l+2} = f_{n-l}\bigl(x_1^{l+1}, \ldots, x_{n-l-1}^{l+1}, \neg x, x_{n-l+1}^{l}, \ldots, x_n^{l}\bigr).$$

Since f_{n-l} is increasing monotonic on input x_{n-l}, $x_{n-l}^{l+2} = x_{n-l}^{l+1} = \neg x$. □

A previous restricted result of Tchuente must be noticed here. He proved [15] that, in a Boolean network with a graph where the only circuits are the loops, and such that $\forall (i, j) \in A$, $i \geqslant j$, we have identical parallel and sequential dynamics. Also, if there exist non-positive loops, we can have cycles as attractors.

The following two lemmas are quite technical, and their purpose is to make the proof of Theorem 5 more straightforward. Lemma 3 states that we can impose values to the last components of the state vector (thus turning them into constants), either before or after applying the operator S, and the result obtained will be the same. Lemma 4 introduces notation for some sequences, and proves their identity.

Lemma 3. *Let N_F be a Boolean network such that $F : \{0,1\}^n \to \{0,1\}^n$ and $\mathcal{S}(N_F) = N_{F^{(1)}}$. If we consider a Boolean network $N_{k,\vec{y}} = N_{F_k}$, $k < n$, such that $F_k : \{0,1\}^k \to \{0,1\}^k$, and, $F_k(x_1, \ldots, x_k) = F(x_1, \ldots, x_k, y_1, \ldots, y_{n-k})$, then $\mathcal{S}(N_{k,\vec{y}}) = N_{F_k^{(1)}}$ where $F_k^{(1)}(x_1, \ldots, x_k) = F^{(1)}(x_1, \ldots, x_k, y_1, \ldots, y_{n-k})$.*

Proof. Let denote by $f_i^{1,k}$ the i-th component of the function $F_k^{(1)}$ and by f_i^k the i-th component of the function F_k. We prove by induction on i that $f_i^{1,k}(\vec{x}) = f_i^{(1)}(\vec{x}, \vec{y})$ for all $x \in \{0,1\}^k$ and for all $i = 1, \ldots, n$.

Basis.

$$f_1^{1,k}(\vec{x}) = f_1^k(\vec{x}) = f_1(\vec{x}, \vec{y}) = f_1^{(1)}(\vec{x}, \vec{y}).$$

Induction hypothesis. $f_l^{1,k}(\vec{x}) = f_l^{(1)}(\vec{x}, \vec{y})$ for all $l = 1, \ldots, i$.

Case $i+1$.

$$f_{i+1}^{1,k}(\vec{x}) = f_{i+1}^k\bigl(f_1^{1,k}(\vec{x}), \ldots, f_i^{1,k}(\vec{x}), x_{i+1}, \ldots, x_k\bigr)$$
$$= f_{i+1}\bigl(f_1^{1,k}(\vec{x}), \ldots, f_i^{1,k}(\vec{x}), x_{i+1}, \ldots, x_k, y_1, \ldots, y_{n-k}\bigr).$$

By induction hypothesis,

$$f_{i+1}^{1,k}(\vec{x}) = f_{i+1}\bigl(f_1^{(1)}(\vec{x}, \vec{y}), \ldots, f_i^{(1)}(\vec{x}, \vec{y}), x_{i+1}, \ldots, x_k, y_1, \ldots, y_{n-k}\bigr)$$
$$= f_{i+1}^{(1)}(\vec{x}, \vec{y}).$$

Then, $F_k^{(1)}(x_1, \ldots, x_k) = F^{(1)}(x_1, \ldots, x_k, y_1, \ldots, y_{n-k})$. □

Lemma 4. *Let N be a Boolean network where the loops, if they exist, are positive. If $N^i = N_{F^{(i)}}$ is the i-sequential network of N and for all $k \leqslant i$, $f_k^{(k)}(x) = f_k^{(k-1)}(x)$ then the sequences:*

1. $X(i,x)^0 = (x_1, \dots, x_i, x_{i+1}, \dots, x_n)$,
   $X(i,x)^{k+1} = (f_1^{(i+1)}(X(i,x)^k), \dots, f_i^{(i+1)}(X(i,x)^k), x_{i+1}, \dots, x_n)$,
2. $Y(i,x)^0 = (x_1, \dots, x_i, x_{i+1}, \dots, x_n)$,
   $Y(i,x)^{k+1} = (f_1^{(i+1-k)}(Y(i,x)^k), \dots, f_i^{(i+1-k)}(Y(i,x)^k), x_{i+1}, \dots, x_n)$,
3. $Z(i,x)^0 = (x_1, \dots, x_i, x_{i+1}, \dots, x_n)$,
   $Z(i,x)^{k+1} = (f_1^{(i-k)}(Z(i,x)^k), \dots, f_i^{(i-k)}(Z(i,x)^k), x_{i+1}, \dots, x_n)$

are identical.

Proof. Notice that by Lemma 3 the sequence $X(i,x)$ is the update of a Boolean network with i variable states, and hence the hypothesis of Lemma 2 applies. Then the sequence converges to a fixed point in i updates, i.e., $X(i,x)^i = X(i,x)^{i+1}$.

In this proof we denote: $X(i,x)^k = x^k$, $Y(i,x)^k = y^k$, $Z(i,x)^k = z^k$. And, we notice that by hypothesis of the lemma:

1. $x^0 = (x_1, \dots, x_i, x_{i+1}, \dots, x_n)$,
   $x^{k+1} = (f_1^{(0)}(x^k), f_1^{(1)}(x^k), \dots, f_i^{(i-1)}(x^k), x_{i+1}, \dots, x_n)$,
2. $y^0 = (x_1, \dots, x_i, x_{i+1}, \dots, x_n)$,
   $y^{k+1} = (f_1^{(0)}(y^k), \dots, f_{i+2-k}^{(i+1-k)}(y^k), \dots, f_i^{(i+1-k)}(y^k), x_{i+1}, \dots, x_n)$,
3. $z^0 = (x_1, \dots, x_i, x_{i+1}, \dots, x_n)$,
   $z^{k+1} = (f_1^{(0)}(z^k), \dots, f_{i+1-k}^{(i-k)}(z^k), \dots, f_i^{(i-k)}(z^k), x_{i+1}, \dots, x_n)$.

Now, we proceed by induction on k.

Basis. Clearly $x^0 = y^0 = z^0$.

Induction hypothesis. For all $j \leqslant k$

$$x^j = y^j = z^j.$$

Case $k+1$. We will only give the proof of sequence x is equal to sequence z, the proof of sequence y is equal to sequence x is analogous.

It is easy to see that:

$$x_l^{k+1} = z_l^{k+1}, \quad \forall l \leqslant i+1-k.$$

Now, if $l > i+1-k \Leftrightarrow k-1 > i-l$:

$$z_l^{k+1} = f_l^{(i-k)}\big(z^k\big).$$

Since $z^k = (f_1^{(i+1-k)}(z^{k-1}), \dots, f_i^{(i+1-k)}(z^{k-1}), x_{i+1}, \dots, x_n)$

$$\begin{aligned} z_l^{k+1} &= f_l^{(i-k)}\big(f_1^{(i+1-k)}\big(z^{k-1}\big), \dots, f_i^{(i+1-k)}\big(z^{k-1}\big), x_{i+1}, \dots, x_n\big) \\ &= f_l^{(i-k)}\big(f_1^{(i+1-k)}\big(z^{k-1}\big), \dots, f_{l-1}^{(i+1-k)}\big(z^{k-1}\big), f_l^{(i+1-k)}\big(z^{k-1}\big), \dots, \\ &\qquad f_i^{(i+1-k)}\big(z^{k-1}\big), x_{i+1}, \dots, x_n\big) \\ &= f_l^{(i-k)}\big(f_1^{(i+1-k)}\big(z^{k-1}\big), \dots, f_{l-1}^{(i+1-k)}\big(z^{k-1}\big), z_l^k, \dots, z_i^k, x_{i+1}, \dots, x_n\big) \end{aligned}$$

by induction hypothesis:

$$z_l^{k+1} = f_l^{(i-k)}\big(f_1^{(i+1-k)}\big(z^{k-1}\big), \ldots, f_{l-1}^{(i+1-k)}\big(z^{k-1}\big), x_l^k, \ldots, x_i^k, x_{i+1}, \ldots, x_n\big).$$

Since $k-1 > i-l$, by Lemma 2:

$$z_l^{k+1} = f_l^{(i-k)}\big(f_1^{(i+1-k)}\big(z^{k-1}\big), \ldots, f_{l-1}^{(i+1-k)}\big(z^{k-1}\big), x_l^{k-1}, \ldots, x_i^{k-1}, x_{i+1}, \ldots, x_n\big),$$

by induction hypothesis:

$$z_l^{k+1} = f_l^{(i-k)}\big(f_1^{(i+1-k)}\big(z^{k-1}\big), \ldots, f_{l-1}^{(i+1-k)}\big(z^{k-1}\big), z_l^{k-1}, \ldots, z_i^{k-1}, x_{i+1}, \ldots, x_n\big),$$
$$z_l^{k+1} = f_l^{(i+1-k)}\big(z^{k-1}\big) = z_l^k.$$

Then by induction hypothesis and Lemma 2:

$$z_l^{k+1} = x_l^k = x_l^{k+1}. \qquad \square$$

4. Algorithm

In this section we present an algorithm, polynomial in the number of vertices of the graph associated to the Boolean network, which allows to "filter" the dynamical cycles of a Boolean network without negative circuits. Additionally, this algorithm allows to find a fixed point in polynomial time. We note that for arbitrary Boolean networks, determining the existence of fixed points and finding one are both strong NP-complete problems [17]. In networks where the associated graph has no negative circuits, we can say that the dependence of a variable on itself is always positive, and thus the existence of fixed points is guaranteed by Theorem 6 in [2].

```
Filter(N)
{
   while (N ≠ S(N))
      N ← S(N)
   return(N)
}
```

The next theorem proves that this algorithm stops in at most n iterations and that it gives a Boolean network without cycles where the fixed points are reached in at most n updates.

Theorem 5. *Let N_F be a Boolean network where $F : \{0,1\}^n \to \{0,1\}^n$ such that all the circuits are positive. Let $N^{n-1} = \underbrace{\mathcal{S} \circ \cdots \circ \mathcal{S}}_{n-1 \text{ times}}(N)$. Then $N^{n-1} = \mathcal{S}(N^{n-1})$, and the only attractors of this network are fixed points, that are reached in at most n updates.*

Proof. We only need to prove that:

$$\forall F : \{0,1\}^n \to \{0,1\}^n, \quad \forall i = 1, \ldots, n, \qquad f_i^{(i)}(x) = f_i^{(i-1)}(x)$$

because the fact that the only attractors of this network are fixed points, that are reached in at most n updates, is given by Lemma 2.

We proceed by induction on i; we will prove that:

$$f_i^{(i)}(x) = f_i^{(i-1)}(x).$$

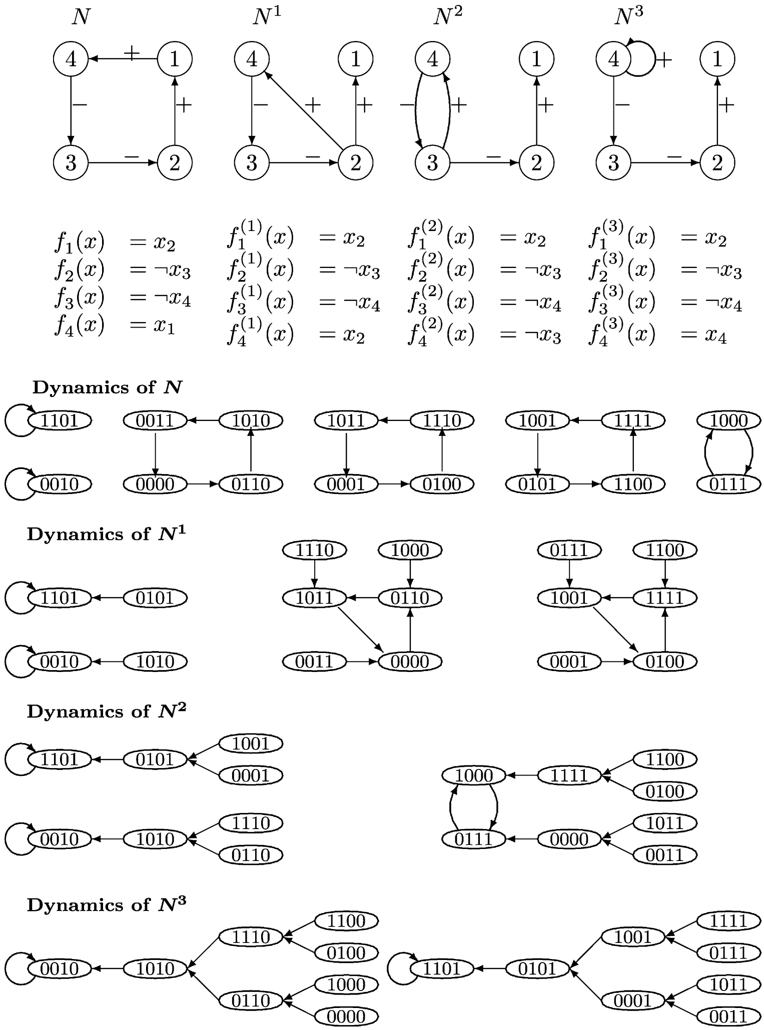


Fig. 2. Algorithm applied on a particular Boolean network.

Basis. For $i = 1$

$$f_1^{(1)}(x) = f_1^{(0)}(x) = f_1(x).$$

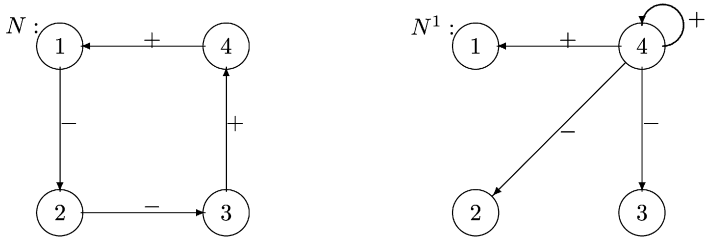
Induction hypothesis. For all $j \leqslant i$, and for all $x \in \{0, 1\}^n$

$$f_j^{(j)}(x) = f_j^{(j-1)}(x).$$

Case $i + 1$. If we consider the sequences defined in Lemma 4, we obtain:

$$f_{i+1}^{(i+1)}(x) = f_{i+1}^{(i)}\big(f_1^{(i+1)}(x), \ldots, f_i^{(i+1)}(x), x_{i+1}, \ldots, x_n\big) = f_{i+1}^{(i)}\big(Y(i,x)^1\big),$$

$$f_{i+1}^{(i+1)}(x) = f_{i+1}^{(i-1)}\big(f_1^{(i)}\big(Y(i,x)^1\big), \ldots, f_i^{(i)}\big(Y(i,x)^1\big), x_{i+1}, \ldots, x_n\big) = f_{i+1}^{(i-1)}\big(Y(i,x)^2\big),$$



$$\begin{aligned} f_1(x) &= x_4 \\ f_2(x) &= \neg x_1 \\ f_3(x) &= \neg x_2 \\ f_4(x) &= x_3 \end{aligned} \qquad \begin{aligned} f_1^{(1)}(x) &= x_4 \\ f_2^{(1)}(x) &= \neg x_4 \\ f_3^{(1)}(x) &= x_4 \\ f_4^{(1)}(x) &= x_4 \end{aligned}$$

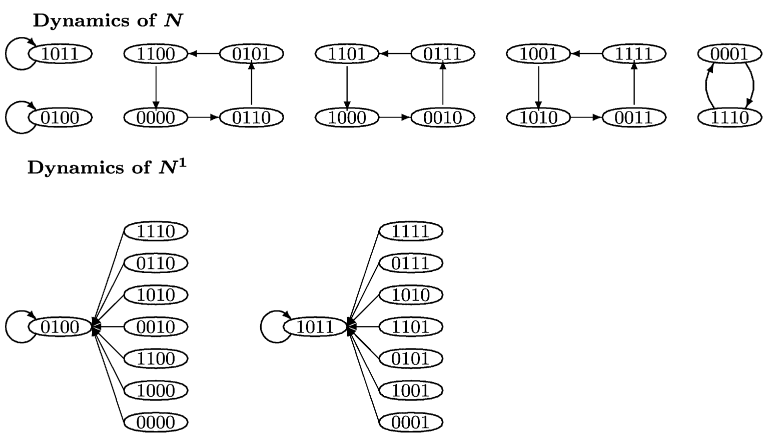


Fig. 3. Algorithm applied on a particular Boolean network.

$$\vdots$$

$$f_{i+1}^{(i+1)}(x) = f_{i+1}\big(Y(i,x)^{i+1}\big),$$

also,

$$f_{i+1}^{(i)}(x) = f_{i+1}^{(i-1)}\big(f_1^{(i)}(x), \ldots, f_i^{(i)}(x), x_{i+1}, \ldots, x_n\big) = f_{i+1}^{(i-1)}\big(Z(i,x)^1\big),$$

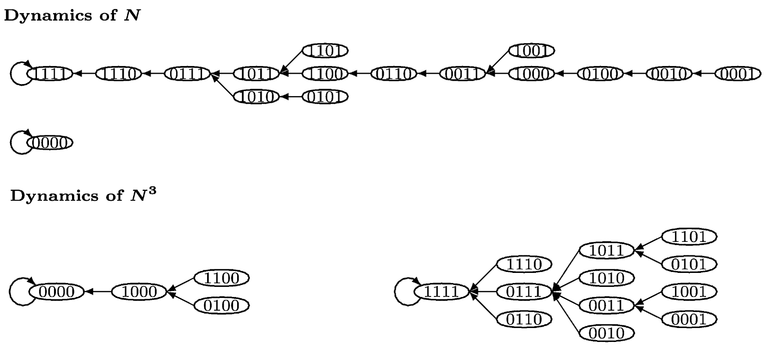
$$f_{i+1}^{(i)}(x) = f_{i+1}^{(i-2)}\big(f_1^{(i-1)}\big(Z(i,x)^1\big), \ldots, f_i^{(i-1)}\big(Z(i,x)^1\big), x_{i+1}, \ldots, x_n\big) = f_{i+1}^{(i-2)}\big(Z(i,x)^2\big),$$

$$\vdots$$

$$f_{i+1}^{(i)}(x) = f_{i+1}\big(Z(i,x)^i\big).$$

By induction hypothesis $f_j^{(j)}(x) = f_j^{(j-1)}(x)$, $\forall j \leqslant i$, then, by Lemma 4:

$$Y(i,x)^{i+1} = X(i,x)^{i+1}, \qquad Z(i,x)^i = X(i,x)^i$$



Functions

N :		N^3 :	
f_1	$= x_2$	$f_1^{(3)}$	$= x_2$
f_2	$= x_3$	$f_2^{(3)}$	$= x_3$
f_3	$= x_1 \vee x_4$	$f_3^{(3)}$	$= x_3 \vee x_4$
f_4	$= x_1$	$f_4^{(3)}$	$= x_3 \vee x_4$

Fig. 4. Boolean network without cycles but with a long transient.

and, by Lemmas 2 and 3, $X(i,x)^{i+1} = X(i,x)^i$. Thus,

$$f_{i+1}^{(i+1)}(x) = f_{i+1}^{(i)}(x). \qquad \square$$

Fig. 2 shows the algorithm applied on a particular Boolean network. We observe that the algorithm converges in $n-1$ time steps; however, we can find in this case an alternative sequential update schedule that allows the algorithm to converge faster (see Fig. 3).

5. Discussion

In first place, it is well known that the problem of finding a fixed point, in the general case, is strong NP-complete [17], but in this case we can found one fixed point in polynomial time. In fact, we note that in the i-th step of the algorithm `Filter` we modify only $n-i$ functions and thus the algorithm is $O(n^2)$. Since in the $(n-1)$-sequential network the fixed points are reached in at most n updates, we can find a fixed point of the Boolean network in $O(n^2)$. Furthermore, if we have a Boolean network N without cycles, the algorithm allows to find the fixed point with shorter transient (see Fig. 4).

The worst case for the algorithm is to converge in n time steps, but if we use an appropriate update sequential schedule the algorithm converges faster. For example, Fig. 3 shows a Boolean network that converges in one iteration of the algorithm. A good strategy to accelerate the convergence is to choose an update schedule that minimizes the number of edges $(i,j) \in A$ such that $i \geqslant j$. In fact, we observe that if a circuit has only one arc (i,j) such that $i \geqslant j$, this circuit becomes a loop in the vertex i and an arc from i to every vertex in the circuit in one step, as shown in Fig. 5. Also we see that using such an update schedule the algorithm gives us a Boolean network where the fixed points are reached in 1 time step.

Then an open problem is to found a polynomial algorithm, if there exists, that allows to get the convergence of the algorithm in one time step and such that the Boolean network obtained reaches the fixed point in one time step. Also, we need to note that, since the algorithm `Filter` converges in $O(n^2)$, this new algorithm would need a lower order of convergence to be useful.

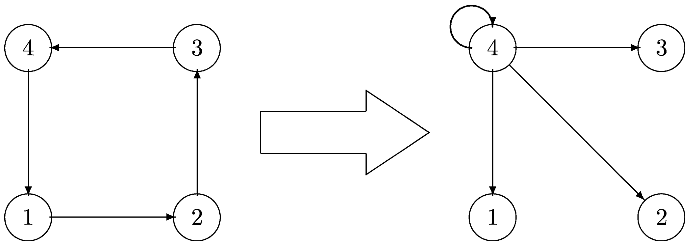


Fig. 5. Circuit that converges in one iteration of the algorithm.

Acknowledgments

The authors would like to thank Andrés Moreira for carefully reading the manuscript and suggesting helpful improvements.

References

[1] M. Anthony, Discrete Mathematics of Neural Networks: Selected Topics, Monographs on Discrete Mathematics and Applications, Society for Industrial and Applied Mathematics, 1975.
[2] J. Aracena, Maximum number of fixed points in regulatory Boolean networks, Bull. Math. Biol. 70 (5) (2008) 1398–1409.
[3] J. Aracena, Modelos matemáticos discretos asociados a los sistemas biológicos. aplicación a las redes de regulación génica, Ph.D. thesis, U. Chile & UJF, Santiago, Chile & Grenoble, Francia, 2001.
[4] J. Aracena, M. González, A. Zúñiga, M. Méndez, V. Cambiazo, Regulatory network for cell shape changes during Drosophila ventral furrow formation, J. Theoret. Biol. 239 (2006) 49–62.
[5] P. Dunne, The Complexity of Boolean Networks, Academic Press, CA, 1988.
[6] S. Harris, B. Sawhill, A. Wuensche, S. Kauffman, A model of transcriptional regulatory networks based on biases in the observed regulation rules, Complexity 7 (2002) 23–40.
[7] F. Jacob, J. Monod, On the regulation of gene activity, Cold Spring Harb. Symp. Quant. Biol. 26 (1961) 193–211.
[8] S. Kauffman, Metabolic stability and epigenesis in randomly connected nets, J. Theoret. Biol. 22 (1969) 437–467.
[9] S. Kauffman, C. Peterson, B. Samuelsson, C. Troein, Random Boolean network models and the yeast transcriptional network, Proc. Nat. Acad. Sci. 100 (2003) 14796–14799.
[10] L. Mendoza, E. Alvarez-Buylla, Dynamics of the genetic regulatory network for Arabidopsis Thaliana flower morphogenesis, J. Theoret. Biol. 193 (1998) 307–319.
[11] S. Nikolajewa, M. Friedel, T. Wilhelm, Boolean networks with biologically relevant rules show ordered behavior, BioSystems 90 (1) (2007) 40–47.
[12] F. Robert, Discrete Iterations, Springer Series in Computational Mathematics, Springer, 1986.
[13] L. Sánchez, D. Thieffry, A logical analysis of the Drosophila Gap-gene system, J. Theoret. Biol. 211 (2001) 115–141.
[14] Z. Szallasi, S. Liang, Modeling the normal and neoplastic cell cycle with “realistic Boolean genetic networks”: Their application for understanding carcinogenesis and assessing therapeutic strategies, in: Proceedings of Pacific Symposium on Biocomputing, vol. 3, 1998, pp. 66–76.
[15] M. Tchuente, Cycles generated by sequential iterations, Discrete Appl. Math. 20 (1988) 165–172.
[16] R. Tocci, R. Widmer, Digital Systems: Principles and Applications, 8th edition, Prentice Hall, NJ, 2001.
[17] Q. Zhao, A remark on “Scalar Equations for Synchronous Boolean Networks with Biological Applications” by C. Farrow, J. Heidel, J. Maloney and J. Rogers, IEEE Trans. Neural Netw. 16 (6) (2005) 1715–1716.