Fault detection problems for Boolean networks and Boolean control networks

FORNASINI Ettore[1], VALCHER Maria Elena[1]

1. Dipartimento di Ingegneria dell'Informazione, Università di Padova, via Gradenigo 6/B, 35131 Padova, Italy
E-mail: fornasini, meme@dei.unipd.it

Abstract: In this paper we address two fault detection problems for Boolean control networks (BCNs). We assume that the BCN may exhibit only two possible configurations, a non-faulty and a faulty one. The fault is simply described as the switching from the non-faulty configuration to the faulty one, and we assume that the BCN cannot autonomously recover from the fault, unless some external intervention restores the regular working conditions. Finally, we suppose that the fault affects only the state-update, not the output measurements. In this set-up, we introduce the concepts of meaningful fault and of detectable meaningful fault. Two different situations are investigated: the case when fault detection must be performed on-line, under arbitrary working conditions, and hence corresponding to arbitrary inputs acting on the BCN, and the case when an off-line test is performed, by making use of a specific input, in order to test whether the BCN is non-faulty or faulty. Complete characterizations and an algorithm to practically perform the tests in both cases are presented. The obtained results for on-line fault detection are finally particularized to the special case of Boolean networks (BNs).

Key Words: Boolean networks, Boolean control networks, Fault detection and identification.

1 Introduction

The current interest in Boolean control networks (BCNs) is undoubtedly motivated by the large number of physical processes whose logical/qualitative behavior can be conveniently described by means of this class of models. Indeed, in a number of contexts, ranging from biology, to game theory and to multi-agent systems, the describing variables display only two operation levels (on/off, high/low, 1/0, ...), and the status of each of them is related to the statuses of the others by means of logical functions (combinations of "and", "or" and "negation" operators). This is the case, in particular, for gene regulatory networks [13, 19], since genes may be treated as binary devices that can be either active or inactive. Also, genes can be activated or inhibited, and this action can be modeled by resorting to external Boolean inputs.

The algebraic representation recently introduced by D. Cheng and co-authors has allowed to cast BCNs into the framework of linear state models (operating on canonical vectors) [1–3]. This new set-up has proved to be extremely beneficial for the research in the field, since it has allowed to derive matrix based characterizations for a number of properties of BCNs, and hence has suggested new approaches to the solution of several control problems for these networks. To mention a few, stability, stabilizability, controllability [15], observability [5] and optimal control [7, 14], have been successfully investigated by referring to the algebraic representations of BCNs.

Research on fault detection originated in the seventies and still represents a lively research area (see [10, 12] for two extended surveys). Fault detection of logic circuits, in particular, has received a lot of attention [11]. Recently, in [17], this problem has been investigated by resorting to the semi-tensor product method. However, the class of logic networks considered in the paper was not described by a BCN and the only faults were "stuck-at faults", resulting in the fact that one (or more) of the input or output variables remains stuck at a certain value. The fault detection problem for gene regulatory networks, described by means of Boolean networks, has been investigated to address some biomedical problems. In [16] it is observed that "the study of diseases such as cancer requires the modeling of gene regulations and the loss of control associated with it. The genetic alterations in the system can be modeled using different fault models in the Boolean Network paradigm." Similarly, in [20], the Authors develop a Boolean network to describe the failure of the oxidative stress response.

Aiming to generalize the results obtained in [16, 20] to the broader context of Boolean control networks, in [8] and [9] we have investigated the situation when, as a consequence of a fault, a BCN switches from its original model to a different one. The main question we have tried to answer is the following one. Assuming that the BCN equations are known, but the state is not accessible, how can we decide whether a fault has occurred, by evaluating the BCN output that corresponds to the applied (known, but otherwise arbitrary) control input? In order to answer this question, we have first identified the class of "meaningful faults", which are the only ones we may hope to identify since they alter the state trajectory, and then introduced the concept of detectability of a meaningful fault. Necessary and sufficient for all meaningful faults to be detectable have been provided, and when such conditions are verified, two practical algorithms to detect meaningful faults have been proposed.

In this paper, we introduce for the fault description the same assumptions as in our previous two contributions, and we are still interested in determining conditions under which meaningful faults are detectable. However, differently from [8] and [9], we consider two possible scenarios: the case when fault detection must be performed on-line, under arbitrary working conditions, and hence corresponding to arbitrary inputs acting on the BCN, and the case when an off-line test is performed, by making use of a specific input, in order to test whether the BCN is non-faulty or faulty. On-line fault detection is essentially what was investigated in [8] and [9], and we recall without proofs the main results obtained in this context, and an algorithm to detect the fault occurrence. Off-line fault detection is here investigated for the first time and a complete characterization of its solvability is provided. It turns out that the algorithm proposed for on-line fault detection can be successfully employed also for off-line fault detection. When dealing with Boolean networks (BNs), on the other hand, an off-line fault detection test is meaningless due to the lack of a control input, and hence only on-line fault detection is considered.

The paper is organized as follows: section 2 introduces the algebraic state representation of a BCN and recalls the definitions of meaningful fault and of detectable meaningful fault. Section 3 addresses on-line fault detection, while section 4 provides a novel and thorough analysis of off-line fault detection. Finally, in section 5 fault detection of BNs is investigated. First, necessary and sufficient conditions for all meaningful faults to be detectable are given, and finally necessary and sufficient conditions for all faults to be meaningful are provided.

Notation. Given $k, n \in \mathbb{Z}_+$, with $k \le n$, the symbol $[k, n]$ denotes the set $\{k, k+1, \dots, n\}$. Boolean vectors and matrices take values in $\mathcal{B} := \{0, 1\}$, with the usual operations (sum $\vee$, product $\wedge$ and negation $\bar{\cdot}$). δ_k^i denotes the ith canonical vector of size k, $\mathcal{L}_k$ the set of k-dimensional canonical vectors, and $\mathcal{L}_{k\times n} \subset \mathcal{B}^{k\times n}$ the set of $k \times n$ matrices whose columns are canonical vectors of size k. Any matrix $L \in \mathcal{L}_{k\times n}$ can be represented as a row whose entries are canonical vectors in $\mathcal{L}_k$, namely $L = [\, \delta_k^{i_1} \quad \delta_k^{i_2} \quad \dots \quad \delta_k^{i_n} \,]$, for suitable indices $i_1, i_2, \dots, i_n \in [1, k]$. The ℓth entry of a vector $\mathbf{v}$ is $[\mathbf{v}]_\ell$.

Given a matrix $L \in \mathcal{B}^{k\times k}$ (in particular, $L \in \mathcal{L}_{k\times k}$), we associate with it a *digraph* $\mathcal{D}(L)$, with vertices $1, \dots, k$. There is an arc (j, ℓ) from j to ℓ if and only if the (ℓ, j)th entry of L is unitary. A sequence $j_1 \to j_2 \to \dots \to j_r \to j_{r+1}$ in $\mathcal{D}(L)$ is a *path* of length r from j_1 to j_{r+1} provided that $(j_1, j_2), \dots, (j_r, j_{r+1})$ are arcs of $\mathcal{D}(L)$. A closed path is a *cycle*. A cycle with no repeated vertices is called *elementary*.

There is a bijective correspondence between Boolean variables $X \in \mathcal{B}$ and vectors $\mathbf{x} \in \mathcal{L}_2$, defined by the relationship

$$\mathbf{x} = \begin{bmatrix} X \\ \bar{X} \end{bmatrix}. \tag{1}$$

We introduce the *(left) semi-tensor product* $\ltimes$ between matrices (in particular, vectors) [3]: given $L_1 \in \mathbb{R}^{r_1\times c_1}$ and $L_2 \in \mathbb{R}^{r_2\times c_2}$ (in particular, $L_1 \in \mathcal{L}_{r_1\times c_1}$ and $L_2 \in \mathcal{L}_{r_2\times c_2}$), we set $L_1 \ltimes L_2 := (L_1 \otimes I_{T/c_1})(L_2 \otimes I_{T/r_2}), T :=$ l.c.m.$\{c_1, r_2\}$, where l.c.m. denotes the least common multiple. If $c_1 = r_2$ then $L_1 \ltimes L_2 = L_1 L_2$. So, the semi-tensor product extends the standard matrix product. By resorting to it, we extend (1) into a bijective correspondence between $\mathcal{B}^n$ and $\mathcal{L}_{2^n}$: given $X = [\, X_1 \quad X_2 \quad \dots \quad X_n \,]^\top \in \mathcal{B}^n$, we set

$$\mathbf{x} := \begin{bmatrix} X_1 \\ \bar{X}_1 \end{bmatrix} \ltimes \begin{bmatrix} X_2 \\ \bar{X}_2 \end{bmatrix} \ltimes \dots \ltimes \begin{bmatrix} X_n \\ \bar{X}_n \end{bmatrix}.$$

Given a sequence $(\mathbf{w}(t))_{t\in\mathbb{Z}_+}$, we denote by $(\mathbf{w}(t))|_{[k,n]}$ its restriction to the "discrete window" $[k, n], k, n \in \mathbb{Z}_+, k \le n$. Similarly, given a set of sequences $\mathfrak{B}$, we denote by $\mathfrak{B}|_{[k,n]} := \{(\mathbf{w}(t))|_{[k,n]} : \exists\, (\mathbf{w}(t))_{t\in\mathbb{Z}_+} \in \mathfrak{B}\}$, the restriction of $\mathfrak{B}$ to $[k, n]$.

2 Preliminaries on Boolean Control Networks and fault detection problems

A *Boolean Control Network* (BCN) is described by the following equations

$$\begin{aligned} X(t+1) &= f(X(t), U(t)), \\ Y(t) &= h(X(t)), \qquad t \in \mathbb{Z}_+, \end{aligned} \tag{2}$$

where $X(t)$, $U(t)$ and $Y(t)$ denote the state variable, the input and the output at time t, taking values in $\mathcal{B}^n, \mathcal{B}^m$ and $\mathcal{B}^p$, respectively. f and h are logic functions, i.e. $f : \mathcal{B}^n \times \mathcal{B}^m \to \mathcal{B}^n$ and $h : \mathcal{B}^n \to \mathcal{B}^p$. By resorting to the semi-tensor product $\ltimes$, the BCN (2) can be described as [3]

$$\begin{aligned} \mathbf{x}(t+1) &= L \ltimes \mathbf{u}(t) \ltimes \mathbf{x}(t), \\ \mathbf{y}(t) &= H \ltimes \mathbf{x}(t) = H\mathbf{x}(t), \qquad t \in \mathbb{Z}_+, \end{aligned} \tag{3}$$

where $\mathbf{x}(t) \in \mathcal{L}_N, \mathbf{u}(t) \in \mathcal{L}_M$ and $\mathbf{y}(t) \in \mathcal{L}_P$, with $N := 2^n, M := 2^m$ and $P := 2^p$. $L \in \mathcal{L}_{N\times NM}$ and $H \in \mathcal{L}_{P\times N}$ are matrices whose columns are canonical vectors. For every $\mathbf{u}(t) = \delta_M^j$, we set $L_j := L \ltimes \mathbf{u}(t) \in \mathcal{L}_{N\times N}$.

Given a BCN described as in (3), we want to investigate the problem of determining, from the measurement of its input and output trajectories (but no access to the state variable), whether a fault has affected the BCN functioning or not. The first step toward this direction consists in defining what do we mean by a fault and what may be the outcome of a fault. In this paper we adopt the same set-up adopted in [8, 9], and assume that the BCN may exhibit only two possible configurations, a non-faulty (NF) and a faulty (F) one. Also, the fault affects only the state-update, not the output measurements, and therefore we represent the non-faulty BCN as in (3) and the faulty one as

$$\begin{aligned} \mathbf{x}(t+1) &= L^{(F)} \ltimes \mathbf{u}(t) \ltimes \mathbf{x}(t), \\ \mathbf{y}(t) &= H \ltimes \mathbf{x}(t) = H\mathbf{x}(t), \qquad t \in \mathbb{Z}_+, \end{aligned} \tag{4}$$

for some suitable $L^{(F)} \in \mathcal{L}_{N\times NM}$, and we set $L_j^{(F)} := L^{(F)} \ltimes \delta_M^j$, $j \in [1, M]$. If we introduce the fault signal $(\mathbf{f}(t))_{t\in\mathbb{Z}_+}$, taking values in $\mathcal{L}_2$, and we assume that $\mathbf{f}(t) = \delta_2^1$ corresponds to the non-faulty BCN and $\mathbf{f}(t) = \delta_2^2$ to the faulty one, the overall BCN dynamics becomes

$$\begin{aligned} \mathbf{x}(t+1) &= \tilde{L} \ltimes \mathbf{f}(t) \ltimes \mathbf{u}(t) \ltimes \mathbf{x}(t), \\ \mathbf{y}(t) &= H \ltimes \mathbf{x}(t) = H\mathbf{x}(t), \qquad t \in \mathbb{Z}_+. \end{aligned} \tag{5}$$

where $\tilde{L} := [\, L \quad L^{(F)} \,] \in \mathcal{L}_{N\times 2NM}$. We assume that the BCN cannot autonomously recover from a fault; so once the fault signal switches from δ_2^1 to δ_2^2, it cannot switch back to δ_2^1, and a fault sequence acting at time $\bar{t}$ is described by the step function

$$\mathbf{f}(t) = \begin{cases} \delta_2^1, & \text{for } 0 \le t < \bar{t}; \\ \delta_2^2, & \text{for } t \ge \bar{t}, \end{cases} \tag{6}$$

where $\bar{t} = +\infty$ in case no fault affects the BCN. We want to investigate under what assumptions on the original and the faulty BCNs, we can detect the fault occurrence from the measurement of the input and output sequences generated by the BCN (5). Specifically, we will consider two possible scenarios in which the fault detection problem may arise:

(1) *on-line fault detection*, corresponding to the case when the BCN, undergoing normal working conditions, is subject to an arbitrary input, and we want to understand, based on its input-output behavior, if a fault has occurred; (2) *off-line fault detection*, by this meaning the situation when we perform an ad hoc off-line test on the BCN, to ascertain whether it is faulty or it is working correctly. In other words, we want to detect whether the BCN is faulty or not, by making use of a single "universal" test (namely by applying some finite support input sequence, independent of the initial condition, that is always supposed to be unknown). In this latter case, we assume that the probability that the BCN becomes faulty during the test is zero, and hence the BCN is either working correctly or erroneously during the whole duration of the test. This amounts to assuming that $\bar{t}$ is either 0 or $+\infty$; equivalently, $\mathbf{f}(t), t \in \mathbb{Z}_+$, is either identically equal to δ_2^1 or to δ_2^2.

To better formalize these problems and their solutions, we denote by $\mathbf{x}(t; \mathbf{x}_0, \mathbf{u}(\cdot), \mathbf{f}(\cdot))$ and $\mathbf{y}(t; \mathbf{x}_0, \mathbf{u}(\cdot), \mathbf{f}(\cdot))$, the state and output vectors of the BCN (5) at time t, when it starts from $\mathbf{x}(0) = \mathbf{x}_0$ and the input and fault sequences are $\mathbf{u}(\cdot)$ and $\mathbf{f}(\cdot)$, respectively.

As a preliminary remark, we notice that a fault taking place at time $\bar{t}$, for certain values of $\bar{\mathbf{x}} := \mathbf{x}(\bar{t}) \in \mathcal{L}_N$ and $\mathbf{u}(t), t \geq \bar{t}$, may not reveal itself, independently of the way we choose the output measurements. Indeed, it is possible that the state trajectory generated by the faulty BCN (4) starting from $\bar{\mathbf{x}}$ at $t = \bar{t}$, under the effect of $\mathbf{u}(\cdot)$, coincides with the state trajectory that the non-faulty BCN (3) generates in the same conditions. This is not an unreasonable situation, since it corresponds to the case when the faulty part of the system is not involved in the dynamic evolution and hence the fault cannot be detected. Under this perspective, it is convenient to introduce the concept of *meaningful fault*.

Definition 1 [8, 9] *Given an initial state $\mathbf{x}_0 \in \mathcal{L}_N$ and an input sequence $(\mathbf{u}(t))_{t \in \mathbb{Z}_+}$, a fault sequence $(\mathbf{f}(t))_{t \in \mathbb{Z}_+}$ induces a* meaningful fault *for the BCN (5) if the state trajectory $(\mathbf{x}(t; \mathbf{x}_0, \mathbf{u}(\cdot), \mathbf{f}(\cdot)))_{t \in \mathbb{Z}_+}$, generated by (5) corresponding to $\mathbf{x}_0$, $\mathbf{u}$ and $\mathbf{f}$, is different from the state trajectory $(\mathbf{x}(t; \mathbf{x}_0, \mathbf{u}(\cdot), \delta_2^1))_{t \in \mathbb{Z}_+}$ (which coincides with the one) generated by the non-faulty system (3) corresponding to the same initial condition and input.*

Meaningful fault sequences are the only ones we may hope to detect, by making use of the input and output trajectories, and hence in the following we will restrict our attention to them. Based on the concept of meaningful fault, we can now formalize and solve the on-line and off-line fault detection problems.

3 On-line fault detection

When dealing with on-line fault detection, we have no restrictions on the initial states and on the input sequences applied to the BCN. As a result, the case when a fault is not meaningful may arise and there is no way to avoid this situation. To give an intuition, it would be as if the windscreen wiper would break in a sunny day. There are almost zero chances of realizing it, unless one tests its functioning purposedly. Even restricting our interest to meaningful faults, however, it is clear that the time instant $\bar{t}$ at which the fault occurs does not necessarily coincide with the first time $t_f \geq \bar{t}$ at which the meaningful fault modifies the state trajectory. And even in the most optimistic scenario, there is no way to evaluate $\bar{t}$, only t_f. So, in the following we will focus our attention only on t_f and we will regard it as the practical time instant at which a meaningful fault occurs.

In order to formalize the concept of detectable fault we introduce the *set of the admissible input/output trajectories* of the non-faulty BCN (3), or equivalently of the BCN (5) corresponding to $\mathbf{f}(t) = \delta_2^1, \forall\, t \in \mathbb{Z}_+$:

$$\begin{aligned}\mathfrak{B}_{uy} \;:=\; & \{(\mathbf{u}(t), \mathbf{y}(t))_{t \in \mathbb{Z}_+} : (\mathbf{u}(t))_{t \in \mathbb{Z}_+} \in (\mathcal{L}_M)^{\mathbb{Z}_+}, \text{and} \\ & \exists \mathbf{x}_0 \in \mathcal{L}_N \text{ s.t. } \mathbf{y}(t) = \mathbf{y}(t; \mathbf{x}_0, \mathbf{u}(\cdot), \delta_2^1)\}. \qquad (7)\end{aligned}$$

In other words, $\mathfrak{B}_{uy}$ is the set of all pairs $(\mathbf{u}(t), \mathbf{y}(t))_{t \in \mathbb{Z}_+}$ such that $(\mathbf{y}(t))_{t \in \mathbb{Z}_+}$ is the output trajectory generated by (3) corresponding to some initial state $\mathbf{x}(0) = \mathbf{x}_0 \in \mathcal{L}_N$ and to the input $(\mathbf{u}(t))_{t \in \mathbb{Z}_+}$.

Definition 2 *Given a BCN (5), an initial state $\mathbf{x}_0 \in \mathcal{L}_N$, an input sequence $(\mathbf{u}(t))_{t \in \mathbb{Z}_+}$, and a (meaningful) fault sequence $(\mathbf{f}(t))_{t \in \mathbb{Z}_+}$, we say that the* (meaningful) fault is detectable *if the input/output pair $(\mathbf{u}(t), \mathbf{y}(t; \mathbf{x}_0, \mathbf{u}(\cdot), \mathbf{f}(\cdot))_{t \in \mathbb{Z}_+}$ generated by the BCN (5) does not belong to $\mathfrak{B}_{uy}$.*

To answer the on-line fault detection problem, we define

$$\begin{aligned}X^* \;:=\; & \{\mathbf{x}^* \in \mathcal{L}_N : \exists\, \mathbf{u}^* \in \mathcal{L}_M \text{ s.t.} \\ & L \ltimes \mathbf{u}^* \ltimes \mathbf{x}^* \neq L^{(F)} \ltimes \mathbf{u}^* \ltimes \mathbf{x}^*\}, \qquad (8)\end{aligned}$$

and, for every $\mathbf{x}^* \in X^*$,

$$U^*(\mathbf{x}^*) := \{\mathbf{u}^* \in \mathcal{L}_M : L \ltimes \mathbf{u}^* \ltimes \mathbf{x}^* \neq L^{(F)} \ltimes \mathbf{u}^* \ltimes \mathbf{x}^*\}. \qquad (9)$$

Also, $\mathcal{UY}^*$ denotes the set of input/output trajectories $(\mathbf{u}(t), \mathbf{y}^{(F)}(t)) \in (\mathcal{L}_M \times \mathcal{L}_P)^{\mathbb{Z}_+}$ generated by the faulty BCN (4) corresponding to some $\mathbf{x}(0) = \mathbf{x}^* \in X^*$ and to some input $(\mathbf{u}(t))_{t \in \mathbb{Z}_+}$ with $\mathbf{u}(0) \in U^*(\mathbf{x}^*)$. In other words, we focus on input/output trajectories for which a fault located at $t = 0$ is meaningful and modifies that state evolution starting at $t = 1$ (so, $t_f = 0$).

Proposition 1 [8, 9] *For the BCN (5) the following facts are equivalent:*

i) for every initial condition $\mathbf{x}_0 \in \mathcal{L}_N$ and every input sequence $(\mathbf{u}(t))_{t \in \mathbb{Z}_+}$, every fault that is meaningful (for the specific choice of $\mathbf{x}_0$ and $\mathbf{u}$) is also detectable;

ii)
$$\mathcal{UY}^* \cap \mathfrak{B}_{uy} = \emptyset. \qquad (10)$$

Condition (10) can be checked by resorting to a graph theoretic approach. The idea is to introduce a graph that is able to keep in parallel the state-transitions in the non-faulty BCN and in the faulty one, starting from any pair of states and corresponding to any input sequence. We introduce the *NF-F (non-faulty-faulty) directed graph* $\mathcal{G} = (\mathcal{V}, \mathcal{E}, C_0, C_1)$, where

- The vertex set $\mathcal{V}$ is the set of all pairs of states, namely $\{(\delta_N^i, \delta_N^j) \in \mathcal{L}_N \times \mathcal{L}_N\}$.

- The labeled edge set $\mathcal{E}$ is defined as follows: there is an edge labeled by $\mathbf{u} \in \mathcal{L}_M$ from the pair (δ_N^i, δ_N^j) to the pair (δ_N^h, δ_N^k) if and only if $\delta_N^h = L \ltimes \mathbf{u} \ltimes \delta_N^i$ and $\delta_N^k = L^{(F)} \ltimes \mathbf{u} \ltimes \delta_N^j$. Note that from every pair (δ_N^i, δ_N^j) there are M outgoing arcs, one for each value of the input $\mathbf{u}$. Clearly, two vertices may be connected by arcs with different labels.
- The vertex set is partitioned into 2 classes: C_0 and C_1. A pair (δ_N^i, δ_N^j) belongs to C_1 if $H\delta_N^i = H\delta_N^j$, while it belongs to C_0 if $H\delta_N^i \neq H\delta_N^j$.

Proposition 2 [9] *Given the BCN (5), let* $\mathcal{G} = (\mathcal{V}, \mathcal{E}, C_0, C_1)$ *be the associated NF-F directed graph. All meaningful faults affecting the BCN are detectable if and only if each path in* $\mathcal{G}$ *endowed with the properties:*
P1) it starts from some vertex pair $(\mathbf{x}_0, \mathbf{x}^*) \in \mathcal{L}_N \times X^*$*;*
P2) the first arc of the path (outgoing from $(\mathbf{x}_0, \mathbf{x}^*)$*) is labeled by some* $\mathbf{u}^* \in U^*(\mathbf{x}^*)$*;*
eventually enters the class C_0*.*

Remark 1 *Proposition 2 provides a necessary and sufficient condition for all meaningful faults to be detectable. The existence of a not detectable meaningful fault corresponds, henceforth, to the case when a path can be found, satisfying P1) and P2) but never leaving the class* C_1*. This ensures the existence in* C_1 *of a cycle that can be reached from the pair* $(\mathbf{x}_0, \mathbf{x}^*)$*. Note that the existence of a cycle in* C_1 *is equivalent to the existence of a not detectable fault. However, this fault is not necessarily meaningful, unless it can be reached starting from a pair* $(\mathbf{x}_0, \mathbf{x}^*)$ *satisfying P1) and P2).*

Remark 2 *Proposition 2 provides a way (at least when* N, M *and* P *are not too large) to check whether meaningful faults are always detectable. Indeed, one simply needs to explore in the NF-F directed graph all the paths endowed with properties P1) and P2) and see after how many steps they enter* C_0*. One may wonder how long these paths may be, in the worst case, and hence how heavy is this test from a computational viewpoint. If every path satisfying P1) and P2) eventually enters* C_0*, it cannot encounter the same vertex pair in* C_1 *twice. So its length is upper bounded by the cardinality of* C_1*, and, in the worst case, the maximum number of distinct path (of length* $|C_1|$ *) we have to evaluate in the NF-F graph is upper-bounded by* $|X^*|R(\max_{\mathbf{x}^* \in X^*} |U^*(\mathbf{x}^*)|)M^{|C_1|-1} \ll N^2 M^{|C_1|}$*, where* R *is the cardinality of the largest set of states that correspond to the same output value.*

3.1 On-line fault detection algorithm

In this section we propose an algorithm to perform on-line fault detection. The idea underlying the algorithm is a somewhat classical one (see [10, 12]), since it essentially performs an observer-based fault detection, one of the most popular fault detection techniques.

A way to test whether a fault occurred or not, consists in verifying, at every time τ, whether the set $\mathbf{X}_\tau^{NF}$ of the states that the non-faulty BCN (3) can reach at time τ, under the effect of the input sequence $\mathbf{u}(t), t \in [0, \tau-1]$, meanwhile generating the output $\mathbf{y}(t), t \in [0, \tau]$, is non-empty.

In other words, one starts at time $\tau = 0$ by determining the set $\mathbf{X}_0^{NF}$ of the initial states compatible with $\mathbf{y}(0)$. At $\tau = 1$, one evaluates $\mathbf{X}_1^{NF}$ of the states that are compatible with $\mathbf{y}(1)$ and can be obtained from the states in $\mathbf{X}_0^{NF}$ by applying $\mathbf{u}(0)$. By proceeding in this way, we obtain the sequence of sets $\mathbf{X}_\tau^{NF}$, whose cardinality decreases with τ. If for some τ we have $\mathbf{X}_\tau^{NF} = \emptyset$, a fault has occurred. On the other had, if $\mathbf{X}_\tau^{NF} \neq \emptyset$, the portion of trajectory $(\mathbf{u}(t), \mathbf{y}(t))_{[0,\tau]}$ belongs to $\mathfrak{B}_{uy}|_{[0,\tau]}$, but considering the delay in revealing the fault, we can only ensure that no meaningful fault has affected the system up to time $\tau - D + 1$, for some $D > 0$ (see [9]). We note that at every time τ the indices of the states that are compatible with a given output sample δ_P^j are the indices of the unitary entries of the Boolean vector $H^\top \delta_P^j$. Based on this remark, the algorithm can be formalized as follows:

Algorithm (Algorithm 2 in [9]).

[Initialization] Set $\tau := 0, \mathbf{X}_0^{NF} := \{\delta_N^i : H\delta_N^i = \mathbf{y}(0)\} = \{\delta_N^i : [H^\top \mathbf{y}(0)]_i = 1\}$ and $\mathbf{v}_0 := \sum_{\delta_N^i \in \mathbf{X}_0^{NF}} \delta_N^i = H^\top \mathbf{y}(0)$.

[Recursive step] Set $\tau := \tau + 1$, and

$$\mathbf{X}_\tau^{NF} := \{\delta_N^i : [L \ltimes \mathbf{u}(\tau-1) \ltimes \mathbf{v}_{\tau-1}]_i = 1 \wedge [H^\top \mathbf{y}(\tau)]_i = 1\}.$$

If $\mathbf{X}_\tau^{NF} = \emptyset$ then a fault occurred: STOP. Otherwise set $\mathbf{v}_\tau := \sum_{\delta_N^i \in \mathbf{X}_\tau^{NF}} \delta_N^i$ and repeat the recursive step.

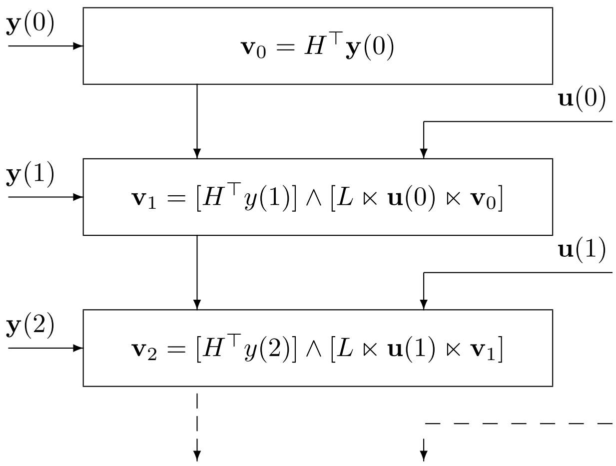

Fig. 1: Flowchart corresponding to the Algorithm.

Example 1 *Consider the BCN (3) with matrices*

$$\begin{aligned} L_1 &:= L \ltimes \delta_2^1 = [\,\delta_4^2 \quad \delta_4^3 \quad \delta_4^4 \quad \delta_4^1\,], \\ L_2 &:= L \ltimes \delta_2^2 = [\,\delta_4^1 \quad \delta_4^3 \quad \delta_4^3 \quad \delta_4^2\,], \\ H &:= [\,\delta_2^1 \quad \delta_2^1 \quad \delta_2^2 \quad \delta_2^2\,], \end{aligned}$$

initialized at $t = 0$ *with* $\mathbf{x}(0) = \delta_4^1$*. Assume that at time* $t = 1$ *a fault occurs and the faulty BCN is described as follows*

$$\begin{aligned} L_1^{(F)} &:= L^{(F)} \ltimes \delta_2^1 = [\,\delta_4^2 \quad \delta_4^3 \quad \delta_4^4 \quad \delta_4^1\,] = L_1, \\ L_2^{(F)} &:= L^{(F)} \ltimes \delta_2^2 = [\,\delta_4^1 \quad \delta_4^3 \quad \delta_4^3 \quad \delta_4^1\,], \end{aligned}$$

(while H *is unaltered). We want to illustrate how the algorithm works when the input sequence is* $\mathbf{u}(0) = \delta_2^1, \mathbf{u}(1) =$

$\delta_2^1, \mathbf{u}(2) = \delta_2^2, \mathbf{u}(3) = \delta_2^1, \mathbf{u}(4) = \delta_2^2, \mathbf{u}(5) = \delta_2^2, ...$ *and the measured output sequence is* $\mathbf{y}(0) = \delta_2^1, \mathbf{y}(1) = \delta_2^1, \mathbf{y}(2) = \delta_2^2, \mathbf{y}(3) = \delta_2^2, \mathbf{y}(4) = \delta_2^2, \mathbf{y}(5) = \delta_2^1, \mathbf{y}(6) = \delta_2^1, ...$
We have:
$X_0^{NF} = \{\delta_4^1, \delta_4^2\}, \mathbf{v}_0 = [1 \quad 1 \quad 0 \quad 0]^\top$.
$X_1^{NF} = \{\delta_4^2\}, \mathbf{v}_1 = [0 \quad 1 \quad 0 \quad 0]^\top$.
$X_2^{NF} = \{\delta_4^3\}, \mathbf{v}_2 = [0 \quad 0 \quad 1 \quad 0]^\top$.
$X_3^{NF} = \{\delta_4^3\}, \mathbf{v}_3 = [0 \quad 0 \quad 1 \quad 0]^\top$.
$X_4^{NF} = \{\delta_4^4\}, \mathbf{v}_4 = [0 \quad 0 \quad 0 \quad 1]^\top$.
$X_5^{NF} = \{\delta_4^2\}, \mathbf{v}_5 = [0 \quad 1 \quad 0 \quad 0]^\top$.
$X_6^{NF} = \emptyset$. *So, we have detected the fault occurrence.*

4 Off-line fault detection

In this section we investigate the problem of determining under what conditions on the BCN (5) it is possible to design a test (equivalently, a time $T > 0$ and a finite input sequence $\hat{\mathbf{u}}(t), t \in [0, T-1]$) such that, independently of the initial condition of the system, we are able to determine from the corresponding output whether the BCN is faulty or non-faulty. As previously said, we assume that the fault cannot happen during the test time, and hence either the BCN is described as in (3) or as in (4) for the whole duration of the test.

The following result provides a straightforward mathematical formalization of the conditions under which the off-line fault detection problem is solvable.

Proposition 3 *For the BCN (5) the following facts are equivalent:*
i) the off-line fault detection problem is solvable;
ii) there exist $T \in \mathbb{Z}_+$ *and an input* $\hat{\mathbf{u}}(t), t \in [0, T-1]$, *taking values in* $\mathcal{L}_M$, *such that the two sets of output trajectories*

$$\hat{\mathcal{Y}}|_{[0,T]} := \{(\mathbf{y}(t))|_{[0,t]} : \exists\, \mathbf{x}_0 \in \mathcal{L}_N \text{ s.t. } \mathbf{y}(t) = \mathbf{y}(t; \mathbf{x}_0, \hat{\mathbf{u}}(\cdot), \delta_2^1), \forall\, t \in [0, T]\} \quad (11)$$

$$\hat{\mathcal{Y}}^{(F)}|_{[0,T]} := \{(\mathbf{y}(t))|_{[0,t]} : \exists\, \mathbf{x}_0 \in \mathcal{L}_N \text{ s.t. } \mathbf{y}(t) = \mathbf{y}(t; \mathbf{x}_0, \hat{\mathbf{u}}(\cdot), \delta_2^2), \forall\, t \in [0, T]\} \quad (12)$$

are disjoint.

The previous proposition suggests an immediate algebraic condition for the solvability of the off-line fault detection problem. Indeed, introduce the *observability matrices in* $T+1$ *steps* [5] of the non-faulty and the faulty BCNs, (3) and (4) respectively, corresponding to the input $\hat{\mathbf{u}}(\cdot)$:

$$\mathcal{O}_{\hat{\mathbf{u}},T+1} := \begin{bmatrix} H \\ HL_{i_0} \\ HL_{i_1}L_{i_0} \\ \vdots \\ HL_{i_{T-1}} \dots L_{i_1}L_{i_0} \end{bmatrix}, \quad (13)$$

$$\mathcal{O}^{(F)}_{\hat{\mathbf{u}},T+1} := \begin{bmatrix} H \\ HL^{(F)}_{i_0} \\ HL^{(F)}_{i_1}L^{(F)}_{i_0} \\ \vdots \\ HL^{(F)}_{i_{T-1}} \dots L^{(F)}_{i_1}L^{(F)}_{i_0} \end{bmatrix}, \quad (14)$$

where we have assumed

$$\hat{\mathbf{u}}(0) = \delta_M^{i_0}, \quad \hat{\mathbf{u}}(1) = \delta_M^{i_1}, \quad \dots \quad , \hat{\mathbf{u}}(T-1) = \delta_M^{i_{T-1}}.$$

The off-line fault detection problem is solvable if and only if there exist $T \in \mathbb{Z}_+$ and indices $i_0, i_1, \dots, i_{T-1} \in [1, M]$ such that the corresponding observability matrices $\mathcal{O}_{\hat{\mathbf{u}},T+1}$ and $\mathcal{O}^{(F)}_{\hat{\mathbf{u}},T+1}$ have no common columns. Obviously, from a computational viewpoint this characterization is not very meaningful, since it would require to evaluate the observability matrices of the two BCNs corresponding to all possible input sequences.

An alternative approach to the problem solution is in terms of the NF-F graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, C_0, C_1)$, previously introduced. Condition ii) in Proposition 3 is equivalent to saying that there exist $T \in \mathbb{Z}_+$ and input values $\delta_M^{i_t}, t \in [0, T-1]$, such that all paths of length T in $\mathcal{G}$ that start from vertices in C_1 and whose ℓth arc is labeled by $\delta_M^{i_{\ell-1}}, \ell \in [1, T]$, pass through some vertex of C_0. The fact that subsequently the path may remain in C_0 or not is irrelevant. By entering in C_0 the output trajectory of the faulty BCN has already proved to be different from the output trajectory of the non-faulty BCN with which we are comparing it.

When N and M are not very large, the graph characterization just provided allows to easily verify whether the off-line fault detection problem is solvable or not, just by direct inspection of the graph $\mathcal{G}$. When N and/or M are large, however, this is not a feasible solution. However, the graph characterization just provided suggests a very simple way to check the problem solvability.

Proposition 4 *For the BCN (5) the following facts are equivalent:*
i) the off-line fault detection problem is solvable;
ii) for every vertex $v = (\delta_N^i, \delta_N^j) \in C_1$ *there is path in the NF-F graph* $\mathcal{G} = (\mathcal{V}, \mathcal{E}, C_0, C_1)$ *from* v *to some vertex belonging to* C_0.

PROOF. i) $\Rightarrow$ ii) follows immediately from the previous reasoning.

ii) $\Rightarrow$ i) Let v_1 be a vertex of C_1. By assumption, there exist $T_1 \in \mathbb{Z}_+, T_1 > 0$, and $\mathbf{u}(t), t \in [0, T_1 - 1]$, such that the path γ_1 of length T_1, starting from v_1 and whose ℓth arc is labeled by $\mathbf{u}(\ell - 1)$, ends in C_0. Let C_2 be the set of distinct vertices in C_1 that are reached in T_1 steps from some vertex $v \in C_1, v \neq v_1$, by means of a path having the same sequence of labelled arcs as the path γ_1 and not entering C_0. In other words, all the vertices along such paths must belong to C_1. Clearly, $|C_2| < |C_1|$. Now consider a vertex $v_2 \in C_2$. By assumption, there exist $T_2 \in \mathbb{Z}_+, T_2 > 0$, and $\mathbf{u}(t), t \in [T_1, T_1 + T_2 - 1]$, such that the path γ_2 of length T_2, starting from v_2 and whose ℓth arc is labeled by $\mathbf{u}(T_1 + \ell - 1)$, ends in C_0. Let C_3 be the set of distinct vertices in C_2 that are reached in T_2 steps from some vertex $v \in C_2, v \neq v_2$, by means of a path having the same sequence of labelled arcs as the path γ_2 and not entering C_0. Again, $|C_3| < |C_2|$. So, by proceeding in this way, we have constructed a finite input sequence $\mathbf{u}(t), t \in [0, T-1]$, such that all paths of length T in $\mathcal{G}$ that start from vertices in C_1 and whose ℓth arc is labeled by $\mathbf{u}(\ell+1), \ell \in [0, T-1]$, pass through some vertex of C_0. This proves that the off-line fault detection problem is solvable. □

Condition ii) of Proposition 4 is not only easy to check on the NF-F graph, but also admits a direct algebraic equivalent. Introduce the new state variable $\mathbf{z}(t) := \mathbf{x}_{NF}(t) \ltimes \mathbf{x}_F(t)$. Clearly, $\mathbf{z}(t) \in \mathcal{L}_{N^2}$ and every value of this canonical vector uniquely determines the values of the two canonical vectors $\mathbf{x}_{NF}(t)$ and $\mathbf{x}_F(t)$, and hence a unique vertex in the NF-F graph. A unique matrix $\Phi \in \mathcal{L}_{N^2 \times N^2 M}$, $\Phi = [\,\Phi_1 \quad \Phi_2 \quad \dots \quad \Phi_M\,]$, can be found such that the *NF-F BCN*

$$\mathbf{z}(t+1) = \Phi \ltimes \mathbf{u}(t) \ltimes \mathbf{z}(t) \tag{15}$$

describes the simultaneous dynamics of the non-faulty and the faulty BCNs, starting from any pair of initial conditions and corresponding to a common input sequence $\mathbf{u}(\cdot)$. Clearly, the NF-F BCN (15) provides the algebraic description of the dynamics of the NF-F graph. By suitably adjusting the characterization of reachability given in [6], it is easy to see that once we define the index sets[1]

$$\begin{aligned} I_0 &:= \{i \in [1, N^2] : \delta^i_{N^2} = \delta^h_N \ltimes \delta^k_N, \exists(\delta^h_N, \delta^k_N) \in C_0\}, \\ I_1 &:= \{i \in [1, N^2] : \delta^i_{N^2} = \delta^h_N \ltimes \delta^k_N, \exists(\delta^h_N, \delta^k_N) \in C_1\}, \end{aligned}$$

and the matrix

$$\Omega := \bigvee_{i=0}^{N^2-1} (\Phi_1 \vee \Phi_2 \vee \dots \vee \Phi_M)^i,$$

condition ii) of Proposition 4 holds if and only if for every $j \in I_1$ there exists $i \in I_0$ such that $[\Omega]_{ij} = 1$.

Remark 3 *Proposition 4 and its algebraic equivalent provide handy ways to check whether the off-line fault detection problem is solvable or not. However, the selection of the specific input sequence to use as test function is still computationally demanding. Greedy heuristic algorithms to find any such input sequence can be found, based on branch and bound techniques, but their performances can vary significantly, based on the structure and the complexity of the BCN (5).*

Remark 4 *Once the input sequence to use as test function has been obtained, the Algorithm described in subsection 3.1 still represents the most immediate way to perform the fault detection test on the given BCN.*

5 Fault detection of Boolean networks

As a special case of the previous analysis, we now consider the fault detection problem for Boolean networks. In this case, there is a non-faulty BN:

$$\begin{aligned} \mathbf{x}(t+1) &= L\mathbf{x}(t), \\ \mathbf{y}(t) &= H\mathbf{x}(t), \qquad t \in \mathbb{Z}_+, \end{aligned} \tag{16}$$

and a faulty BN

$$\begin{aligned} \mathbf{x}(t+1) &= L^{(F)}\mathbf{x}(t), \\ \mathbf{y}(t) &= H\mathbf{x}(t), \qquad t \in \mathbb{Z}_+, \end{aligned} \tag{17}$$

[1] Note that $I_0 \cap I_1 = \emptyset$ and $I_o \cup I_1 = [1, N^2]$.

and the overall BN dynamics can be described in compact form as follows

$$\begin{aligned} \mathbf{x}(t+1) &= \tilde{L} \ltimes \mathbf{f}(t) \ltimes \mathbf{x}(t), \\ \mathbf{y}(t) &= H\mathbf{x}(t), \qquad t \in \mathbb{Z}_+. \end{aligned} \tag{18}$$

where $\tilde{L} = [\,L \quad L^{(F)}\,] \in \mathcal{L}_{N \times 2N}$.

The definition of meaningful fault is a straightforward adaption of Definition 1.

Definition 3 *Given a BN (18) and an initial state $\mathbf{x}_0 \in \mathcal{L}_N$, a fault sequence $(\mathbf{f}(t))_{t\in\mathbb{Z}_+}$, given as in (6), describes a* meaningful fault *if the state trajectory generated by the BN (18) corresponding to $\mathbf{x}_0$ and $\mathbf{f}$ is different from the state trajectory generated by the non-faulty BN (16) corresponding to the same initial condition.*

When dealing with BNs, the distinction between on-line and off-line tests does not make much sense, since we cannot apply some special input to evaluate the BN condition. It is reasonable to see the fault detection problem for a BN always as an on-line evaluation of the BN correct functioning. The definition of detectable meaningful fault is a straightforward adaption of the one given in Definition 2 for BCNs, and it requires the introduction of the *set of admissible output trajectories* of the non-faulty BN (16), equivalently, of the BN (18) corresponding to $\mathbf{f}(t) = \delta^1_2, \forall\, t \in \mathbb{Z}_+$:

$$\begin{aligned} \mathfrak{B}_y := \{&(\mathbf{y}(t))_{t\in\mathbb{Z}_+} : \exists \mathbf{x}_0 \in \mathcal{L}_N \text{ s.t.} \\ &\mathbf{y}(t) = \mathbf{y}(t; \mathbf{x}_0, \mathbf{u}(\cdot), \delta^1_2)\}. \end{aligned} \tag{19}$$

Definition 4 *Given a BN (18), an initial state $\mathbf{x}_0 \in \mathcal{L}_N$, and a (meaningful) fault sequence $(\mathbf{f}(t))_{t\in\mathbb{Z}_+}$, we say that the* (meaningful) fault is detectable *if the output trajectory $(\mathbf{y}(t; \mathbf{x}_0, \mathbf{f}(\cdot)))_{t\in\mathbb{Z}_+}$ generated by the BN (18) does not belong to $\mathfrak{B}_y$.*

The equivalent conditions for the detectability of the meaningful faults from the output trajectories of the BN (18) are given in the following proposition. One of them refers to $\mathfrak{B}_y$ and the sets

$$\begin{aligned} X^* &:= \{\mathbf{x}^* \in \mathcal{L}_N : L\mathbf{x}^* \neq L^{(F)}\mathbf{x}^*\}; \\ \mathcal{Y}^* &:= \{(\mathbf{y}(t))_{t\in\mathbb{Z}_+} : \exists \mathbf{x}_0 \in X^* \text{ s.t.} \\ &\qquad \mathbf{y}(t) = \mathbf{y}(t; \mathbf{x}_0, \mathbf{u}(\cdot), \delta^2_2)\}. \end{aligned}$$

The other refers to the NF-F directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, C_0, C_1)$, associated with the BN, which is an obvious adaptation of the NF-F directed graph associated with a BCN (it corresponds to the case $M = 1$).

Proposition 5 *Given a BN (18), the following facts are equivalent:*

i) for every initial condition $\mathbf{x}_0 \in \mathcal{L}_N$, every fault that is meaningful for the given initial condition is detectable from the output trajectory;

ii) $\mathcal{Y}^* \cap \mathfrak{B}_y = \emptyset$;

iii) in the NF-F directed graph associated with the BN all paths starting from some vertex pair $(\mathbf{x}_0, \mathbf{x}^*)$, with $\mathbf{x}_0 \in \mathcal{L}_N, \mathbf{x}^* \in X^*$, eventually enter C_0.

If any of the previous equivalent conditions holds, (a straightforward adaption of) the Algorithm described in subsection 3.1 allows to detect the fault occurrence.

To conclude the analysis of Boolean networks, it is worth investigating under what conditions all faults are meaningful, by this meaning that for every choice of the initial condition, every fault sequence is meaningful and hence generates a state evolution that is different from the one the non-faulty BN would generate corresponding to the same initial condition. The following characterization refers to the graph structure of the non-faulty BN. Indeed, it is known [5] that every BN has a finite set of (limit) cycles (cycles of length 1 represent equilibrium points of the BN) and every state of the BN reaches a cycle in a finite number of steps. Cycles correspond to the periodic state trajectories of the BN and in turn induce periodic output trajectories [6].

Proposition 6 *Given a BN (18), all faults affecting the BN are meaningful if and only if in every cycle of the non-faulty BN (16) there exists some state δ_N^i belonging to X^*.*

PROOF. [Sufficiency] If the condition on the cycles of the BN (16) holds, then for every $\mathbf{x}_0 \in \mathcal{L}_N$ there exists $i \in [1, N]$ such that the state δ_N^i belongs to X^* and to a cycle, and is reached from $\mathbf{x}_0$ in say $k, k+d, k+2d, k+3d, ...$ steps for suitable integers $k \geq 0$ and $d \geq 1$. So, for every fault $\mathbf{f}(\cdot)$ described as in (6), acting on the BN at time $\bar{t}$, there exists $\ell \in \mathbb{Z}_+$ such that $k + \ell d \geq \bar{t}$, and this ensures that the state trajectory associated with the faulty sequence $\mathbf{f}(\cdot)$ is different from the state trajectory generated by the non-faulty BN corresponding to the same initial condition.

[Necessity] Suppose, by contradiction, that there exists a cycle of the BN (16) with no state belonging to X^*. Then, clearly, for every $\mathbf{x}_0$ belonging to such a cycle and for every fault sequence $\mathbf{f}(\cdot)$, the state trajectory starting in $\mathbf{x}_0$ and associated with the faulty sequence $\mathbf{f}(\cdot)$ coincides with the state trajectory generated by the non-faulty BN corresponding to the same initial condition. $\square$

It is worth remarking that the case when all faults are meaningful could be explored also for BCNs, but it would introduce extremely restrictive conditions on the difference between the matrices L and $L^{(F)}$ that are not realistic to impose. Also, when all faults are meaningful, there is a quite interesting characterization of the condition under which they are detectable.

Proposition 7 *Given a BN (18), if all faults affecting the BN are meaningful, then every such fault is detectable if and only if the non-faulty BN and the faulty BN have no common periodic output trajectory.*

PROOF. Suppose that the non-faulty BN and the faulty BN have a common periodic output trajectory. This means that there exist two (possibly identical) states $\mathbf{x}_0, \bar{\mathbf{x}}_0 \in \mathcal{L}_N$ such that $\mathbf{y}(t; \mathbf{x}_0, \delta_2^1) = \mathbf{y}(t; \bar{\mathbf{x}}_0, \delta_2^2)$ for every $t \in \mathbb{Z}_+$. As all faults affecting the BN are meaningful, $\mathbf{f}(t) = \delta_2^2, \forall\, t \in \mathbb{Z}_+$, is meaningful for $\bar{\mathbf{x}}_0$. This means that there exists $\bar{t} \geq 0$ such that $\mathbf{x}(\bar{t}; \bar{\mathbf{x}}_0, \mathbf{u}(\cdot), \delta_2^2) \in X^*$. But this obviously implies that $\mathcal{Y}^* \cap \mathfrak{B}_y \neq \emptyset$.

Conversely, assume that there exists a meaningful fault that is not detectable, and hence $\exists\ \mathbf{y}(\cdot) \in \mathcal{Y}^* \cap \mathfrak{B}_y \neq \emptyset$. This means that there exists $\mathbf{x}^* \in X^*$ and $\mathbf{x}_0 \in \mathcal{L}_N$ such that $\mathbf{y}(t) = \mathbf{y}(t; \mathbf{x}^*, \delta_2^2) = \mathbf{y}(t; \mathbf{x}_0, \delta_2^1)$ for every $t \in \mathbb{Z}_+$. As the state and hence output trajectories of a BN are eventually periodic, this means that there exists $\bar{t} \in \mathbb{Z}_+$ such that $\mathbf{y}(t; \mathbf{x}_0, \delta_2^1)$ is periodic for every $t \in \mathbb{Z}_+, t \geq \bar{t}$. But then $\bar{\mathbf{y}}(t) := \mathbf{y}(t + \bar{t}; \mathbf{x}_0, \delta_2^1), t \in \mathbb{Z}_+$, is a common periodic trajectory of the non-faulty BN and the faulty BN. $\square$

6 Conclusions

In this paper we have investigated the on-line and off-line fault detection problems for Boolean control networks, by assuming that a BCN may exhibit only two possible configurations, a non-faulty and a faulty one, and that a fault corresponds to a (non-reversible) switching from the non-faulty configuration to the faulty one. Complete characterizations and an algorithm to practically perform the tests in both cases have been presented. The results for the on-line fault detection problem have been particularized to the special case of Boolean networks, and the fault detection problem in the special case when all faults affecting the BN are meaningful has been solved. Future research efforts will aim at investigating to what extent we may identify the exact time t_f at which a fault affecting the BCN (or the BN) has become meaningful, and at considering more complex set-ups, with different possible faults and hence different faulty configurations to be detected and identified.

References

[1] D. Cheng. Input-state approach to Boolean Networks. *IEEE Trans. Neural Networks*, 20 (3):512 – 521, 2009.

[2] D. Cheng and H. Qi. State-space analysis of Boolean Networks. *IEEE Trans. Neural Networks*, 21 (4):584 – 594, 2010.

[3] D. Cheng, H. Qi, and Z. Li. *Analysis and control of Boolean networks*. Springer-Verlag, London, 2011.

[4] X. Ding and P.M. Frank. Fault detection via factorization approach. *Systems & Control Letters*, 14:431–436, 1990.

[5] E. Fornasini and M. E. Valcher. Observability, reconstructibility and state observers of Boolean control networks. *IEEE Tran. Aut. Contr.*, 58 (6):1390 – 1401, 2013.

[6] E. Fornasini and M. E. Valcher. On the periodic trajectories of Boolean Control Networks. *Automatica*, 49:1506–1509, 2013.

[7] E. Fornasini and M.E. Valcher. Optimal control of Boolean control networks. *IEEE Trans. Aut. Contr.*, 59 (5): 1258–1270, 2014.

[8] E. Fornasini and M.E. Valcher. Fault detection of Boolean control networks. *Proc. of the 53rd Conference on Decision and Control (CDC2014)*, pp. 6542–6547, Los Angeles (CA), 2014.

[9] E. Fornasini and M.E. Valcher. Fault Detection Analysis of Boolean Control Networks. to appear in *IEEE Trans. Aut. Contr.*, 60, 2015.

[10] P.M. Frank. Fault diagnosis in dynamic systems using analytical and knowledge based redundancy- A survey and some new results. *Automatica*, 26: 459–474, 1990.

[11] H. Fujiwara and S. Toida. The complexity of fault detection problems for combinational logic circuits. *IEEE Trans. Compu.*, 31(6):555–560, 1982.

[12] I. Hwang, S. Kim, Y. Kim, and C. Eng Seah A Survey of Fault Detection, Isolation, and Reconfiguration Methods. *IEEE Trans. Contr. Sys. Tech.*, 18 (3):636–653, 2010.

[13] S.A. Kauffman. Metabolic stability and epigenesis in randomly constructed genetic nets. *J. Theoretical Biology*, 22:437–467, 1969.

[14] D. Laschov and M. Margaliot. A maximum principle for single-input BCNs. *IEEE Trans. Aut. Contr.*, 56 (4):913–917, 2011.

[15] D. Laschov and M. Margaliot. Controllability of BCNs via the Perron-Frobenius theory. *Automatica*, 48:1218–1223, 2012.

[16] R. Layek and A. Datta. Fault detection and intervention in biological feedback networks. *J. Biol. Syst.*, 20:DOI: 10.1142/S0218339012400062, 2012.

[17] H. Li and Y. Wang. Boolean derivative calculation with application to fault detection of combinational circuits via the semi-tensor product method. *Automatica*, 48 (4):688–693, 2012.

[18] H. Li and Y. Wang. Output feedback stabilization control design for Boolean control networks. *Automatica*, 49:3641–3645, 2013.

[19] I. Shmulevich, E.R. Dougherty, S. Kim, and W. Zhang. From Boolean to probabilistic Boolean networks as models of genetic regulatory networks. *Proc. IEEE*, 90 (11):1778–1792, 2002.

[20] S. Sridharan, R. Layek, A.Datta, and J. Venkatraj. Boolean modeling and fault diagnosis in oxidative stress response. *BMC Genomics*, 13, suppl. 6: S4:DOI:10.1186/1471–2164–13–S6–S4PMCID: PMC3481480, 2012.