Observability and reconstructibility of Probabilistic Boolean Networks

Ettore Fornasini and Maria Elena Valcher

***Abstract*—In this paper observability and reconstructibility properties of Probabilistic Boolean Networks (PBNs) on a finite time interval are addressed. By assuming that the state update follows a probabilistic rule, while the output is a deterministic function of the state, we investigate under what conditions the knowledge of the output measurements in $[0, T]$ allows the exact identification either of the initial state or of the final state of the PBN. By making use of the algebraic approach to PBNs, the concepts of observability, weak reconstructibility and strong reconstructibility are introduced and characterized. Set theoretic algorithms to determine all possible initial/final states compatible with the given output sequence are provided.**

***Index Terms* - Probabilistic Boolean Networks, observability, weak/strong reconstrucitbility.**

I. INTRODUCTION

The current wide-spread interest in Boolean (control) networks (BNs/BCNs) and Probabilistic Boolean networks (PBNs), as effective models of gene regulatory networks, must be credited to Stuart Kauffman. In [10], he first noted that cells regulatory genes behave like switches, taking "on/off" status (1/0, respectively). As the switching often obeys some probabilistic law, PBNs are very convenient models to describe the dynamics of genetic networks. This first intuition stimulated a long stream of research, adopting BNs, PBNs and BCNs to model gene regulatory networks (e.g., [2], [6], [9], [16], [20]).

In the last decade, D. Cheng and co-authors proposed an algebraic approach to BNs, BCNs and PBNs [3], [4], [5] that relies on the semi-tensor product and allows to represent logic networks by means of state-space models. Indeed, every state of a finite-state system can be represented by a canonical vector, and logic relationships by means of logic matrices. Consequently, Boolean (control) networks can be represented as discrete-time (bi)linear systems. On the other hand, a probabilistic Boolean network can be viewed as a Boolean network whose state updating structure switches within a finite set of different models. At each time, the model selection, and hence the switching from the current model into the next one, obeys a given probabilistic law. Equivalently, a PBN can be thought of as a BN in which each state has a family of different successors, and the transition probabilities are known. Fundamental issues as controllability, reachability, state feedback stabilization and optimal control of PBNs have been investigated in the literature [12], [14], [15], [17], [18], [19], with special attention for the application of these results in gene regulatory networks. On the other hand, the research on the observability and reconstructibility of PBNs is still at an early stage [8], [11], [22], and we are not aware of any general theory of reconstructibility for PBNs.

Observability and reconstructibility properties for PBNs offer some interesting challenges. When dealing with a BN, if its output is known in some time window $[0, T]$, it is natural to investigate what are the initial/final states compatible with the given output. Accordingly, observability and reconstructibility are defined as the properties to be able to uniquely identify the initial or final state, respectively, for every measured output. For a PBN, the knowledge of an output sequence in $[0, T]$ allows to determine the sets of all initial/final states compatible with the given sequence, and to introduce a probability distribution on these sets. In this perspective, the natural extension of the observability and reconstructibility definitions given for BNs is that of requiring that for every output sequence the support of the probability distribution consists of a single (initial or final) state, and hence the initial or final states can be uniquely determined (i.e., with probability 1) from the output measurements. The aim of this paper is to investigate these concepts of observability and reconstructibility. It is worth remarking that the observability definitions given in [8], [11], [22] are in the same spirit. Indeed, in [8] observability is defined in a weaker form by imposing that no pair of states is undistinguishable in $[0, T]$, thus ruling out that two distinct states generate the same output sequence with probability 1. In [11] a state is said to be observable if there is at least one output sequence that can be associated with that initial state and with no other initial state. Finally, in [22] the authors adopt two definitions of observability: the first one is equivalent to ours, and imposes that for every pair of distinct initial states the corresponding output sequences on a finite window are distinct with probability 1, the second one, that is equivalent to the one investigated in [8], is weaker, and requires that such a probability is not zero. It is worth noticing that all these notions are substantially deterministic, since they impose that certain properties hold either with probability 1 or with nonzero probability. These definitions not only are the natural extensions of the standard definitions given for BNs (see the discussion in the Introduction of [22]), but are the first unavoidable step one has to take to investigate these concepts for logic systems whose transitions follow a probabilistic rule. A weaker notion of observability, called asymptotical observability in distribution (AOD), has been introduced in [22], and it corresponds to the case when, "for any two distinct initial states, the probability of distinguishability tends to one as the observation period tends to infinity." This is an interesting concept, that may be regarded as a benchmark rather than as a property of practical interest, since in reality the evaluation of the initial/final state must be possible in finite time and hence with a finite number of measurements. Other

E. Fornasini and M.E. Valcher are with the Dipartimento di Ingegneria dell'Informazione Università di Padova, via Gradenigo 6B, 35131 Padova, Italy, e-mail: fornasini@dei.unipd.it, meme@dei.unipd.it.

weaker definitions of observability and reconstructibility could be proposed by allowing, for instance, that the probability distributions of the initial/final states have an arbitrarily small distance ε from a distribution having a single state as support. However, this target does not seem easy to achieve until the aforementioned deterministic problems have been completely explored.

In this paper we address the observability and reconstructibility properties for PBNs, by assuming that the state vector updates according to some probabilistic rule, while the output measurements are deterministic functions of the state vector. This set-up, adopted also in [8] and [22], is justified by the fact that in a lot of situations the output measurements are logic combinations of the state variables unaffected by uncertainty, for instance they are simply a subset of the state measurements. On the other hand, the definition and the results presented in this paper for the observability property would require minor modifications if we would assume a probabilistic model also for the output equation. By making use of the algebraic approach to PBNs, introduced by D. Cheng and co-authors (see [5], Chapter 19), we convert the expression of the PBN into algebraic form and we define and investigate for this model the concept of observability. When a PBN is observable, an upper bound on the minimal length of the observation window that allows to uniquely determine the initial state is provided. Two concepts of reconstructibility, called weak reconstructibility and strong reconstructibility, are then introduced. This is motivated by the fact that the probabilistic nature of a PBN makes the problem more articulated than the one for deterministic models. Indeed, for a BN, once the state $\mathbf{x}(T)$ has been determined from the output sequence $\mathbf{y}(0), \mathbf{y}(1), ..., \mathbf{y}(T)$, all the subsequent states $\mathbf{x}(t), t \geq T$, can be uniquely determined, too. For PBNs, on the contrary, the case may occur that, for a given output sequence $\mathbf{y}(t), t \in \mathbb{Z}_+$, the knowledge of $\mathbf{y}(0), \mathbf{y}(1), ..., \mathbf{y}(T)$ allows to uniquely identify $\mathbf{x}(T)$, but the output sequence $\mathbf{y}(0), \mathbf{y}(1), ..., \mathbf{y}(T+1)$ does not allow to uniquely determine $\mathbf{x}(T+1)$. In general, weak reconstructibility does not imply strong reconstructibility, as it can be shown by very simple counterexamples. Set theoretic algorithms that determine the family of all initial or final states compatible with an admissible output sequence are provided. If the PBN is observable in $[0, T]$ the observability algorithm terminates with a singleton, i.e., a set of cardinality 1, in at most T steps. On the other hand, if it is strongly reconstructible in $[0, T]$, the output of the reconstructibility algorithm is a singleton for all $t \geq T$. Algebraic versions of these algorithms, based on elementary Boolean algebra, and hence very easy to implement, are proposed.

Notation. Given two nonnegative integers $k, n \in \mathbb{Z}_+$, with $k \leq n$, the symbol $[k, n]$ denotes the set $\{k, k+1, \dots, n\}$. We consider Boolean vectors and matrices, taking values in $\mathcal{B} = \{0, 1\}$, with the usual logical operations (And $\wedge$, Or $\vee$, Negation $\overline{\cdot}$). δ^i_k denotes the ith canonical vector of size k, $\mathcal{L}_k$ the set of all k-dimensional canonical vectors, and $\mathcal{L}_{k\times n} \subset \mathcal{B}^{k\times n}$ the set of all $k \times n$ matrices whose columns are canonical vectors of size k. A matrix $L \in \mathcal{L}_{k\times n}$ can be represented as $L = [\, \delta^{i_1}_k \quad \delta^{i_2}_k \quad \dots \quad \delta^{i_n}_k \,]$, for suitable indices $i_1, i_2, \dots, i_n \in [1, k]$. $[A]_{\ell j}$ is the (ℓ, j)th entry of the matrix A. There is a bijective correspondence between Boolean variables $X \in \mathcal{B}$ and vectors $\mathbf{x} \in \mathcal{L}_2$, defined by the relationship $\mathbf{x} = \begin{bmatrix} X \\ \overline{X} \end{bmatrix}$.

The *(left) semi-tensor product* $\ltimes$ between matrices (and hence, in particular, vectors) is defined as follows [5]. Given $L_1 \in \mathcal{L}_{r_1\times c_1}$ and $L_2 \in \mathcal{L}_{r_2\times c_2}$, we set

$$L_1 \ltimes L_2 := (L_1 \otimes I_{T/c_1})(L_2 \otimes I_{T/r_2}), \quad T := \text{l.c.m.}\{c_1, r_2\}.$$

The semi-tensor product is an extension of the standard matrix product, by this meaning that if $c_1 = r_2$, then $L_1 \ltimes L_2 = L_1 L_2$. Note that if $\mathbf{x}_1 \in \mathcal{L}_{r_1}$ and $\mathbf{x}_2 \in \mathcal{L}_{r_2}$, then $\mathbf{x}_1 \ltimes \mathbf{x}_2 \in \mathcal{L}_{r_1 r_2}$. By resorting to the semi-tensor product, we can extend the previous correspondence to a bijective correspondence [5] between $\mathcal{B}^n$ and $\mathcal{L}_{2^n}$. Indeed, given $X = [\, X_1 \quad X_2 \quad \dots \quad X_n \,]^\top \in \mathcal{B}^n$, set

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

II. Observability of PBNs

In this paper by a *Probabilistic Boolean Network* (PBN) we mean a discrete-time Boolean system described by the following equations [8], [11]

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

where $X(t)$ and $Y(t)$ denote the n-dimensional state variable and the p-dimensional output variable at time t, taking values in $\mathcal{B}^n$ and $\mathcal{B}^p$, respectively. f and h are (logic) functions, namely maps $f : \mathcal{B}^n \to \mathcal{B}^n$ and $h : \mathcal{B}^n \to \mathcal{B}^p$. We assume that while h is a fixed logic function, at each time $t \in \mathbb{Z}_+$ the logic map f takes values in a set $\{f_1, f_2, \dots, f_M\}$, and we denote by p_i the probability that the logic map f takes the value f_i. Note that such a probability is independent of t and of $X(t)$.

Upon representing the state and the output vectors $X(t)$ and $Y(t)$ by means of their equivalent $\mathbf{x}(t)$ and $\mathbf{y}(t)$ in $\mathcal{L}_N$ and $\mathcal{L}_P$, respectively, where $N := 2^n$ and $P := 2^p$, at each time t the BN (1) corresponding to a specific choice of the pair $(f_i, h), i \in [1, M]$, can be described as [5]

$$\begin{aligned} \mathbf{x}(t+1) &= L_i \ltimes \mathbf{x}(t) = L_i \mathbf{x}(t), \\ \mathbf{y}(t) &= H \ltimes \mathbf{x}(t) = H\mathbf{x}(t), \qquad t \in \mathbb{Z}_+, \end{aligned} \tag{2}$$

where $L_i \in \mathcal{L}_{N\times N}$ and $H \in \mathcal{L}_{P\times N}$ are matrices whose columns are canonical vectors of size N and P, respectively. Therefore the PBN is a switched Boolean Network [13]:

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

where $\sigma(t) \in [1, M]$, and $\Pr\{\sigma(t) = i\} = p_i$.

Definition 1. *Given a PBN (3) and a time instant $T \in \mathbb{Z}_+$, we say that $\mathbf{y}(0), \mathbf{y}(1), \dots, \mathbf{y}(T)$, with $\mathbf{y}(t) \in \mathcal{L}_P, \forall\ t \in [0, T]$, is an* admissible output sequence *if it can be generated in $[0, T]$ by the PBN (3) corresponding to some initial condition*

$\mathbf{x}(0) = \delta_N^i \in \mathcal{L}_N$ *and some switching signal* $\sigma : [0, T-1] \to [1, M]$.

Definition 2. *Given a time instant* $T \in \mathbb{Z}_+$*, the PBN (3) is* observable in $[0, T]$ *if, for every admissible output sequence* $\mathbf{y}(0), \mathbf{y}(1), \dots, \mathbf{y}(T)$*, it is possible to uniquely identify the corresponding initial condition* $\mathbf{x}(0) = \delta_N^i$*. The PBN is* observable *if it is observable in some interval* $[0, T]$.

Remark 1. *Despite a PBN can be thought of as the BCN*

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

with $L := \sum_{i=1}^M p_i L_i$*, driven by the stochastic input sequence* $\mathbf{u}(t) \in \mathcal{L}_M$*,* $\Pr\{\mathbf{u}(t) = \delta_M^i\} = p_i$*, the standard observability problem for BCNs, as stated in [7], [21] is intrinsically different, as the identification of the initial state* $\mathbf{x}(0)$ *depends on the knowledge of both the output sequence* $\mathbf{y}(0), \mathbf{y}(1), \dots, \mathbf{y}(T) \in \mathcal{L}_P$ *and the input sequence* $\mathbf{u}(0), \mathbf{u}(1), \dots, \mathbf{u}(T-1) \in \mathcal{L}_M$*. In the case of a PBN, we assume, on the contrary, that the input is not available, a situation that is reminiscent of the "observability with unknown-inputs problem" (see [1]).*

In order to analyze the observability problem, we introduce the observability matrix $\mathcal{O}_{\sigma,T}$ associated with a specific sequence $\sigma(t), t \in [0, T-1]$:

$$\mathcal{O}_{\sigma,T} := \begin{bmatrix} H \\ HL_{\sigma(0)} \\ HL_{\sigma(1)}L_{\sigma(0)} \\ \vdots \\ HL_{\sigma(T-1)} \dots L_{\sigma(1)}L_{\sigma(0)} \end{bmatrix}. \tag{5}$$

We provide the following result, whose proof is elementary and hence omitted.

Proposition 2. *Given a PBN (3) and a time instant* $T \in \mathbb{Z}_+$*, let* $\sigma_1, \sigma_2, \dots, \sigma_R$*, where* $R = M^T$*, be all the possible distinct sequences* $\sigma(t), t \in [0, T-1]$*, taking values in* $[0, M]$*. The PBN is observable in* $[0, T]$ *if and only if for every choice of the indices* $i, j \in [1, R]$ *and* $h, k \in [1, N]$*, condition* $\mathcal{O}_{\sigma_i,T}\ \delta_N^h = \mathcal{O}_{\sigma_j,T}\ \delta_N^k$ *implies* $h = k$.

If all columns of the matrix

$$\mathcal{O}_T := [\, \mathcal{O}_{\sigma_1,T} \quad \mathcal{O}_{\sigma_2,T} \quad \dots \quad \mathcal{O}_{\sigma_R,T} \,] \tag{6}$$

are distinct, then the PBN is observable in $[0, T]$ (and, in this particular case, one can identify from the output observation also the sequence σ_i).

Remark 3. *If the output update would follow a probabilistic model, too, namely* $\mathbf{y}(t)$ *would be expressed as* $\mathbf{y}(t) = H_{\sigma(t)}\mathbf{x}(t)$*, the previous proposition would still hold, provided that the matrices* $\mathcal{O}_{\sigma_i,T}$ *would be suitably defined by replacing each block* $HL_{\sigma(t-1)} \dots L_{\sigma(1)}L_{\sigma(0)}$ *with* $H_{\sigma(t)}L_{\sigma(t-1)} \dots L_{\sigma(1)}L_{\sigma(0)}$.
In [22], finite-time observability with probability 1 *is proved to be equivalent to a form of set-reachability, and set reachability is characterized in turn by resorting to STG reconstruction. We believe that while STG reconstruction is a valuable means to investigate set reachability, the current characterisation of observability is simpler to derive and check.*

Proposition 4. *A PBN (3) is observable if and only if it is observable in* $[0, N(N+1)/2]$.

Proof. It is clear that if a PBN is observable in $[0, T]$, then it is observable in every interval $[0, T']$ with $T' \geq T$. On the other hand, suppose that $T = \frac{N(N+1)}{2}$ and there exist two initial states $\mathbf{x}_1(0) = \delta_N^i$ and $\mathbf{x}_2(0) = \delta_N^j, i \neq j$, and two sequences $\sigma_1(t), t \in [0, T-1]$, and $\sigma_2(t), t \in [0, T-1]$, such that the corresponding output sequences, say $\mathbf{y}_1(t)$ and $\mathbf{y}_2(t), t \in [0, T]$, coincide at every time instant. Since the set of all unordered pairs $\{a, b\}$ with $a, b \in [1, N]$, and a not necessarily distinct from b, has cardinality[1] $\frac{N(N+1)}{2}$, the number of distinct pairs in $\{\{\mathbf{x}_1(t), \mathbf{x}_2(t)\} : t \in [0, T]\}$ is at most $\frac{N(N+1)}{2}$. Consequently, there exist $0 \leq \tau < \bar{\tau} \leq T$ such that $\{\mathbf{x}_1(\tau), \mathbf{x}_2(\tau)\} = \{\mathbf{x}_1(\bar{\tau}), \mathbf{x}_2(\bar{\tau})\}$. We distinguish two cases:
(1) If $(\mathbf{x}_1(\tau), \mathbf{x}_2(\tau)) = (\mathbf{x}_1(\bar{\tau}), \mathbf{x}_2(\bar{\tau}))$ (and this is the case, in particular, if $\mathbf{x}_1(\tau) = \mathbf{x}_2(\tau)$), then define the two (eventually periodic) switching sequences

$$\bar{\sigma}_1(t) := \begin{cases} \sigma_1(t), & t \in [0, \bar{\tau}-1]; \\ \bar{\sigma}_1(t - \bar{\tau} + \tau), & t \geq \bar{\tau}, \end{cases}$$

$$\bar{\sigma}_2(t) := \begin{cases} \sigma_2(t), & t \in [0, \bar{\tau}-1]; \\ \bar{\sigma}_2(t - \bar{\tau} + \tau), & t \geq \bar{\tau}. \end{cases}$$

(2) If $(\mathbf{x}_1(\tau), \mathbf{x}_2(\tau)) = (\mathbf{x}_2(\bar{\tau}), \mathbf{x}_1(\bar{\tau}))$, then define the two (eventually periodic) switching sequences

$$\bar{\sigma}_1(t) := \begin{cases} \sigma_1(t), & t \in [0, \bar{\tau}-1]; \\ \bar{\sigma}_2(t - \bar{\tau} + \tau), & t \geq \bar{\tau}, \end{cases}$$

$$\bar{\sigma}_2(t) := \begin{cases} \sigma_2(t), & t \in [0, \bar{\tau}-1]; \\ \bar{\sigma}_1(t - \bar{\tau} + \tau), & t \geq \bar{\tau}. \end{cases}$$

In both cases, corresponding to the switching sequences[2] $\bar{\sigma}_1$ and $\bar{\sigma}_2$, the two initial states $\mathbf{x}_1 = \delta_N^i$ and $\mathbf{x}_2 = \delta_N^j, i \neq j$, generate exactly the same output sequence at every time $t \in \mathbb{Z}_+$, and hence the system cannot be observable in any time window $[0, T'], T' \geq T$. □

Note that also Proposition 4 applies to PBNs whose output update follows the probabilistic rule described in Remark 3.
In order to further analyze the observability problem, we provide an algorithm to identify - when possible - the initial state associated with a given output sequence $\mathbf{y}(0), \mathbf{y}(1), \dots, \mathbf{y}(T)$, taking values in $\mathcal{L}_P$. If the output sequence is admissible, the algorithm provides the set $\mathcal{X}_0$ of all initial states compatible with that output sequence and, when the PBN is observable in $[0, T]$, it uniquely identifies the initial state. On the other hand, if the output sequence is not admissible, eventually the set $\mathcal{X}_0$ becomes empty. To introduce the algorithm, we need to

[1]Note that we are assuming that $\{a, b\} = \{b, a\}$, but we are not ruling out the case when $a = b$.

[2]Note that in case (1) the switching sequence is eventually periodic of period $T = \bar{\tau} - \tau$, while in case (2) the period is $T = 2(\bar{\tau} - \tau)$, since the new sequences $\bar{\sigma}_1$ and $\bar{\sigma}_2$ alternate the portions of σ_1 and σ_2 in $[\tau, \bar{\tau} - 1]$.

define (see [7]) the set of states that produce the same output value. Specifically, for every $t \in [0, T]$, we set

$$\mathcal{C}(\mathbf{y}(t)) := \{\delta_N^i \in \mathcal{L}_N : H\delta_N^i = \mathbf{y}(t)\}.$$

We also introduce the set of *one-step predecessors* of a family of states $\mathcal{T} \subset \mathcal{L}_N$:

$$\mathcal{P}(\mathcal{T}) := \{\delta_N^i \in \mathcal{L}_N : \exists h \in [1, M] \text{ such that } L_h\delta_N^i \in \mathcal{T}\}.$$

Clearly, one can inductively define the set of *k-step predecessors* of the family $\mathcal{T}$, $\mathcal{P}^k(\mathcal{T})$, as

$$\mathcal{P}^1(\mathcal{T}) = \mathcal{P}(\mathcal{T}), \qquad \mathcal{P}^k(\mathcal{T}) = \mathcal{P}\left(\mathcal{P}^{k-1}(\mathcal{T})\right).$$

It is worth noting that the previous sets are very easy to compute, since

$$\begin{aligned} \mathcal{C}(\mathbf{y}(t)) &= \{\delta_N^i \in \mathcal{L}_N : \mathbf{y}(t)^\top H\delta_N^i = 1\} \\ &= \{\delta_N^i \in \mathcal{L}_N : [H^\top \mathbf{y}(t)]_i = 1\}, \\ \mathcal{P}^k(\mathcal{T}) &= \{\delta_N^i : [L^k]_{ji} \neq 0, \exists\, j \text{ such that } \delta_N^j \in \mathcal{T}\}. \end{aligned}$$

Algorithm 1 [Determines $\mathcal{X}_0$, the set of initial states compatible with the output sequence $\mathbf{y}(0), \mathbf{y}(1), \dots, \mathbf{y}(T)$]
Input: $\mathbf{y}(0), \mathbf{y}(1), \dots, \mathbf{y}(T)$
Output: $\mathcal{X}_0$
Initialization: $t \leftarrow 0$ and $\mathcal{X}_0 \leftarrow \mathcal{C}(\mathbf{y}(0))$
Loop process: **while** $t \leq T-1$ and $\mathcal{X}_0 \neq \emptyset$ **do**
$t \leftarrow t+1$
$\mathcal{X}_0 \leftarrow \mathcal{X}_0 \cap \mathcal{P}^t(\mathcal{C}(\mathbf{y}(t)))$
end while

Remark 5. *The possible outcomes of the algorithm are three: (1) The algorithm terminates at some time $t \in [0, T]$ with $\mathcal{X}_0 = \emptyset$. If so, the output sequence is not admissible. (2) The algorithm terminates at $t = T$ with $|\mathcal{X}_0| = 1$ and hence it uniquely identifies the initial condition. (3) The algorithm terminates at $t = T$ with $|\mathcal{X}_0| > 1$, providing the set of all initial conditions compatible with the given output sequence.*

The previous algorithm can be easily implemented by resorting to a sequence of algebraic operations involving Boolean matrices and vectors. To this goal we first need to introduce the Boolean sum of the matrices $L_i, i \in [1, M]$, i.e., $L_\mathcal{B} := \vee_{i=1}^M L_i \in \mathcal{B}^{N \times N}$. Also, we introduce the following N-dimensional Boolean vectors:

- $\mathbf{s}(t) := H^\top \mathbf{y}(t), t \in [0, T]$, is the Boolean vector whose ith entry is 1 if and only if $\delta_N^i \in \mathcal{C}(\mathbf{y}(t))$;
- $\mathbf{v}(t)$ is the Boolean vector whose ith entry is 1 if and only if $\mathbf{x}(0) = \delta_N^i$ is an initial state vector compatible with the output sequence $\mathbf{y}(0), \mathbf{y}(1), \dots, \mathbf{y}(t)$. Note that $\mathbf{v}(t)$ represents the set of possible estimates of $\mathbf{x}(0)$ one can make based on the output samples $\mathbf{y}(0), \mathbf{y}(1), \dots, \mathbf{y}(t)$, and hence provides the same information as the set $\mathcal{X}_0$ at step t of Algorithm 1.

Finally, we observe that a state δ_N^j is a one-step predecessor of one of the states δ_N^i such that $[\mathbf{s}(t)]_i = 1$ if and only if the set of unitary entries in the jth column of $L_\mathcal{B}$ is a subset of the set of unitary entries of $\mathbf{s}(t)$. It is not hard to see that the Boolean vector whose unitary entries are those and those only corresponding to the one-step predecessors of the states represented by $\mathbf{s}(t)$ can be expressed as $\overline{[L_\mathcal{B}^\top \overline{\mathbf{s}(t)}]}$. This expression corresponds to first evaluating the (entrywise) negation of vector $\mathbf{s}(t)$, then multiplying it by $L_\mathcal{B}^\top$, and finally calculating the negation of the vector $L_\mathcal{B}^\top \overline{\mathbf{s}(t)}$. Similarly, we can obtain the Boolean vector representing all the k-step predecessors of the states represented by $\mathbf{s}(t)$ as $\overline{[(L_\mathcal{B}^\top)^k \overline{\mathbf{s}(t)}]}$. In light of the previous remarks, Algorithm 1 can be rephrased as follows.

Input: $\mathbf{y}(0), \mathbf{y}(1), \dots, \mathbf{y}(T)$
Output: $\mathbf{v}(T)$
Initialization: $t \leftarrow 0$, $\mathbf{s}(0) \leftarrow H^\top \mathbf{y}(0)$ and $\mathbf{v}(0) \leftarrow \mathbf{s}(0)$
Loop process: **while** $t \leq T-1$ **do**
$t \leftarrow t+1$
$\mathbf{s}(t) \leftarrow H^\top \mathbf{y}(t)$ and $\mathbf{v}(t) \leftarrow \mathbf{v}(t-1) \wedge \overline{[(L_\mathcal{B}^\top)^t \overline{\mathbf{s}(t)}]}$
end while

Example 1. *Consider the PBN (3), with $N = 8$, $P = 4$, $M = 2$, and $L_1 = [\,\delta_8^7\ \delta_8^1\ \delta_8^2\ \delta_8^3\ \delta_8^2\ \delta_8^4\ \delta_8^6\ \delta_8^7\,]$, $L_2 = [\,\delta_8^8\ \delta_8^7\ \delta_8^3\ \delta_8^3\ \delta_8^2\ \delta_8^5\ \delta_8^7\ \delta_8^7\,]$, $H = [\,\delta_4^1\ \delta_4^3\ \delta_4^1\ \delta_4^4\ \delta_4^4\ \delta_4^2\ \delta_4^3\ \delta_4^1\,]$ and $p_1 = p_2 = 1/2$. Applying the algorithm to the admissible output sequence $\mathbf{y}(0) = \delta_4^1$, $\mathbf{y}(1) = \delta_4^1$, $\mathbf{y}(2) = \delta_4^3$, $\mathbf{y}(3) = \delta_4^1$, $\mathbf{y}(4) = \delta_4^3$, one gets:*
$t = 0, \mathbf{s}(0) = \delta_8^1 + \delta_8^3 + \delta_8^8 = \mathbf{v}(0).$
$t = 1, \mathbf{s}(1) = \delta_8^1 + \delta_8^3 + \delta_8^8, \mathbf{v}(1) = \delta_8^1 + \delta_8^3.$
$t = 2, \mathbf{s}(2) = \delta_8^2 + \delta_8^7, \mathbf{v}(2) = \delta_8^1 + \delta_8^3.$
$t = 3, \mathbf{s}(3) = \delta_8^1 + \delta_8^3 + \delta_8^8, \mathbf{v}(3) = \delta_8^3.$
So, the algorithm uniquely identifies the state $\mathbf{x}(0) = \delta_8^3$.

III. RECONSTRUCTIBILITY OF PBNS

When dealing with reconstructibility of PBNs, we need to introduce two distinct notions, a weak one and a strong one. This situation is different not only from what happens with the observability of PBNs, as discussed in the previous section, but also with what happens when dealing with the reconstructibility of BCNs. We start with the weak notion and then comment on the rationale that led us to introduce also a strong one.

Definition 3. *Given a PBN (3) and a time instant $T \in \mathbb{Z}_+$, the PBN is* weakly reconstructible *in $[0, T]$ if for every admissible output sequence $\mathbf{y}(0), \mathbf{y}(1), \dots, \mathbf{y}(T)$ taking values in $\mathcal{L}_P$, there exists $\tau \in [0, T]$ (depending on the specific output sequence) such that the knowledge of the output samples $\mathbf{y}(0), \mathbf{y}(1), \dots, \mathbf{y}(\tau)$ allows to uniquely identify $\mathbf{x}(\tau) \in \mathcal{L}_N$. The PBN is* weakly reconstructible *if it is weakly reconstructible in some interval $[0, T]$.*

Weak reconstructibility is quite a different property with respect to observability. Indeed, due to the stochastic nature of the state transitions, and differently from what it happens with linear state-space models and with BCNs, the fact that one can identify $\mathbf{x}(\tau)$ at a specific time τ, based on the output observation till time τ, does not ensure that when the next output sample is acquired at time $\tau + 1$ we can still uniquely identify $\mathbf{x}(\tau + 1)$. So, based on the specific output sequence, the time instant within $[0,T]$ at which we are sure about the state value may change. On the other hand, due to the time-invariance of the PBN, if weak reconstructibility is possible in some finite window $[0,T]$, then it is possible in every window $[T_1,T_2]$, with $T_2-T_1 \geq T$. This leads to the following result.

Proposition 6. *Given a PBN (3), the following facts are equivalent:*

i) *there exists $T \in \mathbb{Z}_+$, such that the PBN is weakly reconstructible in $[0,T]$;*
ii) *there exists $\bar{T} \in \mathbb{Z}_+$, such that for every admissible output sequence $\mathbf{y}(t), t \in \mathbb{Z}_+$, an infinite sequence of time instants $\{\tau_k\}_{k=0}^{+\infty}$ can be found, with $\tau_0 := 0 \leq \tau_1 < \tau_2 < \ldots$ and $\tau_{k+1}-\tau_k \leq \bar{T}$, such that the state vectors $\mathbf{x}(\tau_k), k \geq 1$, of the PBN are uniquely determined.*

Proof. i) $\Rightarrow$ ii) Suppose that i) holds for some T, and hence for every admissible output sequence $\mathbf{y}(0),\mathbf{y}(1),\ldots,\mathbf{y}(T) \in \mathcal{L}_P$, there exists $\tau_1 \in [0,T]$ such that the knowledge of the output samples $\mathbf{y}(0),\mathbf{y}(1),\ldots,\mathbf{y}(\tau_1)$ allows to uniquely identify $\mathbf{x}(\tau_1) \in \mathcal{L}_N$. But then, by applying the same reasoning to the admissible output sequence $\mathbf{y}(\tau_1+1),\mathbf{y}(\tau_1+2),\ldots,\mathbf{y}(\tau_1+T+1) \in \mathcal{L}_P$, we can find $\tau_2 \in [\tau_1+1,\tau_1+T+1]$ such that the knowledge of the output sequence $\mathbf{y}(\tau_1+1),\mathbf{y}(\tau_1+2),\ldots,\mathbf{y}(\tau_1+T+1) \in \mathcal{L}_P$ allows to uniquely identify $\mathbf{x}(\tau_2)$, with $\tau_2-\tau_1 \leq T+1$, and so on. So, ii) holds for $\bar{T}=T+1$.
ii) $\Rightarrow$ i) is obvious. □

Remark 7. *Reconstructibility of PBNs is different from reconstructibilty of BCNs [7], due to the fact that one needs to "reconstruct" the final state based only on the output sequence, while ignoring the "input" (equivalently, the switching sequence $\sigma(t), t \in \mathbb{Z}_+$).*

We now provide an algorithm to reconstruct - when possible - the final state $\mathbf{x}(\tau)$ associated with a given output sequence $\mathbf{y}(0),\mathbf{y}(1),\ldots,\mathbf{y}(\tau)$, at some time $\tau \in [0,T]$. For the sake of simplicity, we assume that the output sequence is admissible. The algorithm stops either when $\mathbf{x}(\tau)$ is uniquely determined at some time $\tau \in [0,T]$ or when $t=T$ and if so it provides the set $\mathcal{X}_T$ of all final states at $t=T$ compatible with that output. To introduce the algorithm, we need to define the concept of *one-step successors* of a family of states $\mathcal{T} \subset \mathcal{L}_N$:

$$\mathcal{S}(\mathcal{T}) := \{\delta_N^i \in \mathcal{L}_N : \delta_N^i = L_h\delta_N^j,\ \exists h \in [1,M], \exists \delta_N^j \in \mathcal{T}\}.$$

Such a set is easy to determine, since

$$\begin{aligned}\mathcal{S}(\mathcal{T}) &= \{\delta_N^i : [L]_{ij} \neq 0, \exists\, j \text{ such that } \delta_N^j \in \mathcal{T}\}\\ &= \{\delta_N^i : [L_{\mathcal{B}}]_{ij} = 1, \exists\, j \text{ such that } \delta_N^j \in \mathcal{T}\}.\end{aligned}$$

Algorithm 2 [Uniquely identifies $\mathbf{x}(\tau)$ at some time $\tau \in [0,T]$ or, if not possible, determines the set $\mathcal{X}_T$ of final states compatible with the output sequence $\mathbf{y}(0),\mathbf{y}(1),\ldots,\mathbf{y}(T)$]
Input: $\mathbf{y}(0),\mathbf{y}(1),\ldots,\mathbf{y}(T)$
Output: either a canonical vector $\mathbf{x}(\tau)$ at some time $\tau \in [0,T]$ or $\mathcal{X}_T$
Initialization: $t \leftarrow 0$ and $\mathcal{X}_t \leftarrow \mathcal{C}(\mathbf{y}(t))$
Loop process: **while** $t \leq T-1$ and $|\mathcal{X}_t| \neq 1$ **do**
$t \leftarrow t+1$
$\mathcal{X}_t \leftarrow \mathcal{S}(\mathcal{X}_{t-1}) \cap \mathcal{C}(\mathbf{y}(t))$.
end while

Also in this case, we may rewrite Algorithm 2 in algebraic form by making use of the same notation previously adopted to describe Algorithm 1. In this context we use $\mathbf{v}(t)$ to denote the Boolean vector whose ith entry is 1 if and only if $\mathbf{x}(t)=\delta_N^i$ is a state vector at time t compatible with the output sequence $\mathbf{y}(0),\mathbf{y}(1),\ldots,\mathbf{y}(t)$. We note that if $\mathbf{v}(t-1)$ is the Boolean vector whose unitary entries represent the state vectors that at $t-1$ are compatible with $\mathbf{y}(0),\mathbf{y}(1),\ldots,\mathbf{y}(t-1)$, then the Boolean vector whose unitary entries represent the successors of all such state vectors is simply $L_{\mathcal{B}}\mathbf{v}(t-1)$. We then get the following result.

Input: $\mathbf{y}(0),\mathbf{y}(1),\ldots,\mathbf{y}(T)$
Output: either $\mathbf{v}(t) \in \mathcal{L}_N$ for some $t \in [0,T]$ or $\mathbf{v}(T)$
Initialization: $t \leftarrow 0$, $\mathbf{s}(0) \leftarrow H^\top\mathbf{y}(0)$ and $\mathbf{v}(0) \leftarrow \mathbf{s}(0)$
Loop process: **while** $t \leq T-1$ and $\mathbf{v}(t) \notin \mathcal{L}_N$ **do**
$t \leftarrow t+1$
$\mathbf{s}(t) \leftarrow H^\top\mathbf{y}(t)$ and $\mathbf{v}(t) \leftarrow \mathbf{s}(t) \wedge L_{\mathcal{B}}\mathbf{v}(t-1)$
end while

We introduce, now, a strong definition of reconstructibility and then relate it to the weak one, previously defined.

Definition 4. *Given a PBN (3) and a time instant $T \in \mathbb{Z}_+$, we say that the PBN is* strongly reconstructible *in $[0,T]$ if, given any admissible output sequence $\mathbf{y}(0),\mathbf{y}(1),\ldots,\mathbf{y}(T) \in \mathcal{L}_P$, it is possible to uniquely identify $\mathbf{x}(T) \in \mathcal{L}_N$. The PBN (3) is* strongly reconstructible *if it is strongly reconstructible in some interval $[0,T]$.*

Remark 8. *Compared to weak reconstructibility, strong reconstructibility represents the possibility of identifying in a deterministic way all the final states from the output trajectories at* the same time *T (and not within the time window $[0,T]$). This is equivalent to saying that identification of the final state $\mathbf{x}(t)$ from $\mathbf{y}(0),\mathbf{y}(1),\ldots,\mathbf{y}(t) \in \mathcal{L}_P$ is possible for every $t \geq T$, and not just for an infinite sequence of time instants. Indeed, if the knowledge of the output sequence $\mathbf{y}(0),\mathbf{y}(1),\ldots,\mathbf{y}(T) \in \mathcal{L}_P$ allows to uniquely identify $\mathbf{x}(T) \in \mathcal{L}_N$, then the knowledge of $\mathbf{y}(1),\mathbf{y}(2),\ldots,\mathbf{y}(T+1) \in \mathcal{L}_P$ allows to uniquely identify $\mathbf{x}(T+1) \in \mathcal{L}_N$, and so on.*

Clearly, strong reconstructibility requires the existence of a subset $\mathcal{S}$ of $\mathcal{L}_N$ such that, once we know that the state $\mathbf{x}(t)$ belongs to $\mathcal{S}$ and we measure the output sample at $t+1$, $\mathbf{y}(t+1)$, we can uniquely identify the successor $\mathbf{x}(t+1)$ of $\mathbf{x}(t)$ belonging to the class $\mathcal{C}(\mathbf{y}(t+1))$. In other words, there must be a set $\mathcal{S}$ of state vectors whose successors generate distinct outputs. The largest such set is defined as follows

$$\begin{aligned}\hat{\mathcal{S}} \ :=\ &\{\delta_N^i \in \mathcal{L}_N : \forall\, h,k \in [1,M],\ L_h\delta_N^i \neq L_k\delta_N^i\\ &\Rightarrow\ HL_h\delta_N^i \neq HL_k\delta_N^i\}\end{aligned} \tag{7}$$

and it provides the key ingredient that relates weak reconstructibility and strong reconstructibility.

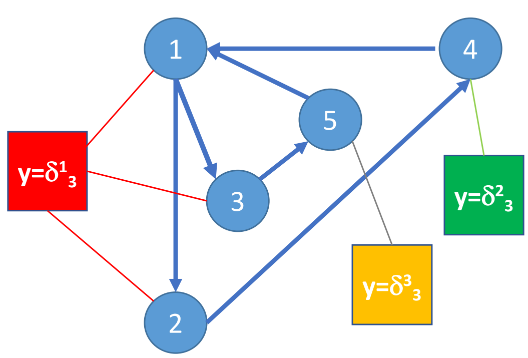
Proposition 9. *Given a PBN (3), the following facts are equivalent:*

i) *the PBN is strongly reconstructible;*

ii) *the PBN is weakly reconstructible and there exists a set* $\mathcal{S} \subseteq \hat{\mathcal{S}}$ *that is* invariant, *by this meaning that if* $\delta_N^i \in \mathcal{S}$ *then for every* $h \in [1, M]$ *the vector* $L_h \delta_N^i$ *belongs to* $\mathcal{S}$ *in turn, and* attractive, *namely for every* $\mathbf{x}(0) \in \mathcal{L}_N$ *and every* $\sigma : \mathbb{Z}_+ \to [1, M]$ *there exists* $\tau \in \mathbb{Z}_+$ *such that* $\mathbf{x}(t) \in \mathcal{S}$ *for every* $t \geq \tau$.

Proof. i) $\Rightarrow$ ii) Strong reconstructibility obviously implies weak reconstructibility, because if the PBN state can always be identified at the time instant T, then, in particular, it can be identified at some time $\tau \in [0, T]$. On the other hand, strong reconstructibility implies (see Remark 8) that there exists $T \in \mathbb{Z}_+$ such that, for every admissible output sequence, it is possible to uniquely identify $\mathbf{x}(t)$ from $\mathbf{y}(0), \mathbf{y}(1), \dots, \mathbf{y}(t) \in \mathcal{L}_P$, for every $t \geq T$. If $\mathbf{x}(T)$ would not be in $\hat{\mathcal{S}}$, then there would be two possible successors of $\mathbf{x}(T)$ generating the same output, but this means that the case may occur that once the sample $\mathbf{y}(T+1)$ is acquired it is not possible to uniquely identify $\mathbf{x}(T+1)$ even if $\mathbf{x}(T)$ is known. On the other hand, the same argument can be used for all the states $\mathbf{x}(t), t \geq T$. But this means that $\mathbf{x}(t) \in \hat{\mathcal{S}}$ for every $t \geq T$. Therefore all such states belong to a subset $\mathcal{S}$ of $\hat{\mathcal{S}}$ that is invariant. Finally, since every state trajectory eventually enters such a set $\mathcal{S}$, this means that $\mathcal{S}$ is attractive.
ii) $\Rightarrow$ i) If the system is weakly reconstructible, then, for every admissible output sequence $\mathbf{y}(t), t \in \mathbb{Z}_+$, there exists a diverging sequence of time instants $\{\tau_k\}_{k=1}^{+\infty}$ such that the knowledge of the output sequence $\mathbf{y}(t), t \in \mathbb{Z}_+$ allows to uniquely identify $\mathbf{x}(\tau_k)$. On the other hand, the attractiveness of $\mathcal{S}$ ensures that there exists $k^* \in \mathbb{Z}_+$ such that $\mathbf{x}(\tau_k) \in \mathcal{S}$ for every $k \geq k^*$. But then $\mathbf{x}(t) \in \mathcal{S}$ for every $t \geq \tau_{k^*}$ and this ensures that the knowledge of $\mathbf{x}(t)$ and $\mathbf{y}(t+1)$ allows to uniquely identify also $\mathbf{x}(t+1)$. Therefore $\mathbf{x}(t)$ is uniquely identified for every $t \geq \tau_{k^*}$. □

A simple PBN that is weakly but not strongly reconstructible is proposed in the following example.



Example 2. *Consider the PBN (3), with* $N = 5$, $P = 3^3$ *and* $M = 2$

$$L_1 = [\,\delta_5^2 \quad \delta_5^4 \quad \delta_5^5 \quad \delta_5^1 \quad \delta_5^1\,],\; L_2 = [\,\delta_5^3 \quad \delta_5^4 \quad \delta_5^5 \quad \delta_5^1 \quad \delta_5^1\,],$$
$$H = [\,\delta_3^1 \quad \delta_3^1 \quad \delta_3^1 \quad \delta_3^2 \quad \delta_3^3\,]$$

[3]Note that the fact that N and P are powers of 2 is irrelevant in our analysis.

and $p_1 = p_2 = 1/2$. *The directed graph corresponding to the PBN is illustrated in Figure 1, below. A circle labelled "i" represents the state vector* δ_N^i, *while the three square boxes represent the three possible output values. Edges connecting circles represent possible state transitions, while an arc from a circle to a square box indicates the output value corresponding to a state vector. It is very easy to verify that the PBN is weakly reconstructible in* $[0, 2]$. *On the other hand it is not strongly reconstructible, since every time the state trajectory moves from* δ_5^1 *to either* δ_5^2 *or to* δ_5^3, *we cannot understand from the output sample in which state we are.*

References

[1] G. Basile and G. Marro. On the observability of linear, time-invariant systems with unknown inputs. *J. Optimization Theory and Appl.*, 3 (6):410–415, 1969.
[2] M. Chaves, M. Albert, and E. D. Sontag. Robustness and fragility of Boolean models for genetic regulatory networks. *J. Theoretical Biology*, 235 (3):431–449, 2005.
[3] D. Cheng. Input-state approach to Boolean Networks. *IEEE Trans. Neural Networks*, 20, (3):512 – 521, 2009.
[4] D. Cheng and H. Qi. Linear representation of dynamics of Boolean Networks. *IEEE Trans. Automatic Control*, 55, (10):2251 – 2258, 2010.
[5] D. Cheng, H. Qi, and Z. Li. *Analysis and control of Boolean networks*. Springer-Verlag, London, 2011.
[6] A. Datta, R. Pal, A. Choudhary, and E.R. Dougherty. Control approaches for probabilistic gene regulatory networks - What approaches have been developed for addressing the issue of intervention? *IEEE Signal Processing Magazine*, 24 (1):54–63, 2007.
[7] E. Fornasini and M. E. Valcher. Observability, reconstructibility and state observers of Boolean control networks. *IEEE Tran. Aut. Contr.*, 58 (6):1390 – 1401, 2013.
[8] Z. Jing and L. Zhenbin. Observability of probabilistic Boolean networks. In *Proc. 2015 34th Chinese Control Conference*, pages 183–186, Hangzhou, China, 2015.
[9] G. Karlebach and R. Shamir. Modelling and analysis of gene regulatory networks. *Nature Reviews Molecular Cell Biology*, 9:770–780, 2008.
[10] S.A. Kauffman. Metabolic stability and epigenesis in randomly constructed genetic nets. *J. Theoretical Biology*, 22:437–467, 1969.
[11] T. Leifeld, Z. Zhang, and P.Zhang. Fault detection for Probabilistic Boolean Networks. In *Proc. 2016 European Control Conference*, pages 740–745, Aalborg, Denmark, 2162.
[12] F. Li and J. Sun. Controllability of probabilistic Boolean control networks. *Automatica*, 47 (12):2765–2771, 2011.
[13] H. Li and Y. Wang. On reachability and controllability of switched Boolean control networks. *Automatica*, 48 (11):2917–22, 2012.
[14] R. Li, M. Yang, and T.G. Chu. State feedback stabilization for probabilistic Boolean networks. *Automatica*, 50:1272–1278, 2014.
[15] Z. Li and H.Xiao. Weak reachability of probabilistic Boolean Control networks. In *Proc. 2015 Int. Conf. Advanced Mechatronic Systems*, pages 56–60, Beijing, China, 2015.
[16] J. Liang and J. Han. Stochastic Boolean networks: An efficient approach to modeling gene regulatory networks. *BMC Systems Biology*, 2016.
[17] Q. Liu. An optimal control approach to probabilistic Boolean networks. *Physica A: Statistical Mechanics and its Applications*, 391 (24):6682–6689, 2012.
[18] Q. Liu. Optimal finite horizon control in gene regulatory networks. *Eur. Phys. J. B*, 86: 245:1–5,DOI: 10.1140/epjb/e2013–30746–7, 2013.
[19] R. Pal, A. Datta, and E. R. Dougherty. Optimal infinite-horizon control for probabilistic boolean networks. *IEEE Trans. Signal Processing,*, 54 (6):2375–2387, 2006.
[20] I. Shmulevich and E.R. Dougherty. *Probabilistic Boolean Networks: The Modeling and Control of Gene Regulatory Networks*. Society for Industrial and Applied Mathematics Philadelphia, PA, 2009.
[21] K. Zhang and L. Zhang. Observability of Boolean Control Networks: A unified approach based on Finite Automata. *EEE Trans. Automatic Control*, 61 (9):2733–2738, 2016.
[22] R. Zhou, Y. Guo, and W. Gui. Set reachability and observability of probabilistic Boolean networks. *Automatica*, 106:230–24, 2019.