

Polynomial-Time Controllability Analysis of Boolean Networks

Koichi Kobayashi, Jun-ichi Imura, and Kunihiro Hiraishi

Abstract—This paper discusses the controllability problem of Boolean networks with inputs (control nodes) and outputs (controlled nodes). An algorithm for testing controllability is in general NP-hard, and the existing polynomial-time algorithm is limited to a class of tree-structure networks. In this paper, based on a sufficient condition for controllability, a polynomial-time algorithm is proposed. The proposed algorithm is applicable to a wider class of large-scale Boolean networks compared with the existing algorithm. The key idea in our approach is to use an adjacency matrix of a directed graph induced by a Boolean network, and Boolean operations are not focused. The effectiveness of the proposed approach is shown by an example of a biological network.

I. INTRODUCTION

Recently, much attention has been paid on modeling, analysis, and control synthesis of biological networks such as gene regulatory networks and metabolic networks in not only theoretical biology community but also control community [1]. Biological networks are in general complicate, and can be expressed as ordinary/partial differential equations with high nonlinearity and high dimensionality. For such a complex system, it will be one of the suitable approaches to use simpler models such as Petri nets, Bayesian networks, Boolean networks, and hybrid systems (see e.g., [7], [9]).

In particular, for a piecewise affine model, which is a kind of hybrid systems, and a Boolean network model, the controllability problem has been developed so far. However, a piecewise affine model can be applied to only the class of relatively low-dimensional systems [3], [5]. A Boolean network model expresses the state as binary variables (inhibit or activate), and the transition rules of the state are given by Boolean functions [10], [11]. So a Boolean network will be more practical for analysis of *large-scale* biological systems thanks to its bold simplification. Thus various approaches based on this model have been well-studied so far (see e.g., [2], [6], [12], [13]). The controllability problem of Boolean networks has been discussed in [4]. This problem is to determine whether expressions of some selected genes (controlled nodes) can be inhibited or activated by expressions of the other gene (control nodes) in a gene regulatory network. In addition, it has been proven in [4] that this problem is NP-hard in a general setting, and, for the classes of networks including a tree structure or at most one loop, a polynomial-time algorithm for controllability test has been proposed. For Boolean network models, there is a criticism that this

model is too simple as a model of biological networks, but as a first step towards further detailed analysis of large-scale biological networks, this model will provide some indications.

In this paper, the controllability problem of Boolean networks with inputs (control nodes) and outputs (controlled nodes) is also considered, and based on simple operations on an adjacency matrix of a directed graph induced by a Boolean network, a sufficient condition for the Boolean network to be controllable is proposed. The proposed condition is given as the form of an algorithm, which is a polynomial-time algorithm. A key in this paper is to give up computing complex Boolean operations in a rigorous way and to focus on deriving an easily-checkable sufficient condition for controllability so as to be applied to large-scale networks. Therefore, the proposed algorithm can be applied to a wider class of Boolean networks including non-tree structures compared with the method in [4].

Notation: Let \mathcal{N} denote the set of nonnegative integers, and $\{0, 1\}^{m \times n}$ the set of $m \times n$ matrices consisting of elements 0 and 1. We also denote by I_n and $0_{m \times n}$ the $n \times n$ identity matrix and the $m \times n$ zero matrix, respectively. For simplicity of notation, we sometimes use the symbol 0 instead of $0_{m \times n}$ and the symbol I instead of I_n . Let M^T express the transpose of the matrix M .

II. BOOLEAN NETWORK MODELS

In this section, a Boolean network model [10], [11] is briefly explained. A Boolean network model consists of a set of nodes and a set of regulation rules for nodes, where each node expresses a gene, a molecule, or an event in the genetic network. The state variable ξ_i at node i takes a Boolean value of 0 or 1 representing the respective “inactive” or “active” status of the node. A regulation rule for each node is given in terms of a Boolean function, and each node state changes synchronously. Thus the dynamics of this network are in general given by the state equation

$$\xi(k+1) = f_a(\xi(k)) \quad (1)$$

where $\xi(k) = [\xi_1(k) \ \xi_2(k) \ \cdots \ \xi_l(k)]^T \in \{0, 1\}^l$ is the state vector at time $k \in \mathcal{N}$, and $f_a : \{0, 1\}^l \rightarrow \{0, 1\}^l$ is a Boolean function consisting of the logical AND (\wedge), OR (\vee), NOT (\neg), and XOR (\oplus).

As an example, we will consider a Boolean network in Fig. 1, which is given in [4]. In this example, the state ξ_1 of node 1 at time $k+1$ is given by the logical AND of the state ξ_2 of node 2 and the state ξ_3 of node 3 at time k . In a similar way, ξ_2 and ξ_3 at time $k+1$ are given by the state ξ_1 of node 1 and the logical NOT of the state ξ_2 of node 2

K. Kobayashi is with Japan Advanced Institute of Science and Technology, Ishikawa 923-1292, Japan k-kobaya@jaist.ac.jp

J. Imura is with Tokyo Institute of Technology, Tokyo 152-8552, Japan imura@mei.titech.ac.jp

K. Hiraishi is with Japan Advanced Institute of Science and Technology, Ishikawa 923-1292, Japan hira@jaist.ac.jp

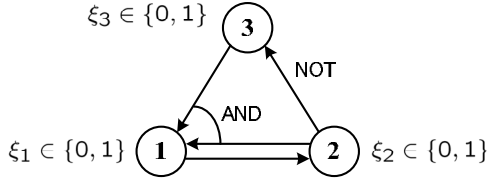


Fig. 1. Example of Boolean network model

at time k , respectively. Thus the Boolean network model of Fig. 1 is expressed as the state equation

$$\begin{cases} \xi_1(k+1) = \xi_2(k) \wedge \xi_3(k), \\ \xi_2(k+1) = \xi_1(k), \\ \xi_3(k+1) = \neg \xi_2(k). \end{cases} \quad (2)$$

For example, if the state at time k is given as $\xi(k) = [1 \ 0 \ 1]^\top$, then the state at time $k+1$ is given as $\xi(k+1) = [0 \ 1 \ 1]^\top$.

III. PROBLEM FORMULATION

A. Boolean network model with inputs/outputs

In a Boolean network model (1), the state $\xi(k)$ is uniquely determined by giving the initial state $\xi(0) = \xi_0 \in \{0,1\}^n$, which implies that (1) is an autonomous system and has no control inputs.

On the other hand, this paper will consider the Boolean network model with control (i.e., input) nodes and controlled (i.e., output) nodes to discuss the output-controllability of this model. This model is given by

$$\Sigma \begin{cases} x(k+1) = f(x(k), u(k)), \\ y(k) = Cx(k) \end{cases} \quad (3)$$

where $x \in \{0,1\}^n$ is the state, $u \in \{0,1\}^m$ is the input, $y \in \{0,1\}^r$ ($r \leq n$) is the output, $f: \{0,1\}^n \times \{0,1\}^m \rightarrow \{0,1\}^n$ is a Boolean function consisting of the logical AND, OR, NOT, and XOR, and $C \in \{0,1\}^{r \times n}$ is the output matrix satisfying for each element c_{ij} of C , $\sum_{i=1}^r c_{ij} = 1, \forall j$ and $\sum_{j=1}^r c_{ij} = 1, \forall i$. Furthermore, the product of C and x in $y = Cx$ expresses a product operation on matrices/vectors of the real number field. Thus the above condition on C guarantees that the output is the state variable itself, i.e., for each i there exists j such that $y_i = x_j$ holds. For example, the case of $y = x$ is also included here. Note that this condition on C will not be restrictive in analyzing controllability of biological networks such as gene regulatory networks, since the relation on regulation among genes/molecules will be discussed there.

We will give an example of the above model. For the Boolean network (2) represented in Fig. 1, consider to choose either $\xi_1(k)$, $\xi_2(k)$ or $\xi_3(k)$ to be the control input. For example, suppose that $x(k) = [\xi_1(k) \ \xi_2(k)]^\top$ and $u(k) = \xi_3(k)$, that is, $\xi_3(k)$ itself is the control input. Then it follows that

$$\begin{cases} x_1(k+1) = x_2(k) \wedge u(k), \\ x_2(k+1) = x_1(k) \end{cases} \quad (4)$$

where $x_i(k)$ denotes the i -th element of $x(k)$. Note here that $\xi_3(k+1) = \neg \xi_2(k)$ is ignored because we assume that

$\xi_3(k)$ itself is the control input. In a similar way, $x(k) = [\xi_1^\top(k) \ \xi_3^\top(k)]^\top$ and $x(k) = [\xi_2^\top(k) \ \xi_3^\top(k)]^\top$ can be considered in this case. As for the output $y = Cx$, either case of $C = I_2$, $C = [1 \ 0]$, $C = [0 \ 1]$ can be considered in this case.

B. Definition of Output-Controllability

For the system Σ of (3), the notion of output-controllability is defined as follows.

Definition 1: Suppose that for the system Σ of (3), the finite time $T \in \mathcal{N}$ and the initial state $x_0 \in \{0,1\}^n$ are given. Then the system Σ is said to be T -output-controllable at x_0 , if for every $y_f \in \{0,1\}^r$ there exists a control input sequence $u(k) \in \{0,1\}^m$, $k = 0, 1, \dots, T-1$, such that $y(T) = y_f$. Furthermore, the system Σ is said to be T -output-controllable if it is T -output-controllable at every x_0 .

The above notion of controllability comes from the fact that in control of genetic networks we often would like to determine if expressions of certain specified gene states will be able to be inhibited (or activated). Note also that the control time T is explicitly specified in the above definition.

As an example, consider whether the Boolean network with inputs and outputs (4) is T -output-controllable or not by directly calculating state trajectories of the system. Consider the controllability of the system (4) with $y = x$ (i.e., $C = I_2$) for $T = 2$. From (4) we have

$$\begin{cases} x_1(2) = x_1(0) \wedge u(1), \\ x_2(2) = x_2(0) \wedge u(0). \end{cases} \quad (5)$$

So if $x_1(0) = 0$, then $x_1(2) \equiv 0$ holds irrespective of the value of $u(1)$. Similarly, if $x_2(0) = 0$, then $x_2(2) \equiv 0$ holds. Therefore we see that the system (4) is not 2-output-controllable. In the same way, we see that the system (4) is not T -output-controllable in every case of $C = I_2$, $C = [1 \ 0]$, and $C = [0 \ 1]$ for $T \geq 2$. On the other hand, for the system (2), suppose that $x(k) = [\xi_1^\top(k) \ \xi_3^\top(k)]^\top$ and $u(k) = \xi_2(k)$. Then we obtain the Boolean network with inputs/outputs

$$\begin{cases} x_1(k+1) = u(k) \wedge x_2(k), \\ x_2(k+1) = \neg u(k) \end{cases} \quad (6)$$

Consider the controllability of this system (6) for $T = 2$. From (6) we have

$$\begin{cases} x_1(2) = u(1) \wedge (\neg u(0)), \\ x_2(2) = \neg u(1). \end{cases} \quad (7)$$

Thus we see that the system is not 2-output-controllable for $C = I_2$, while that the system is T -output-controllable with $T \geq 2$ for both cases of $C = [1 \ 0]$ and $C = [0 \ 1]$.

Note that for (5) with $y = x$ we see that the controllability property does not hold due to the fact that $y(T)$ directly depends on $x(0)$. On the other hand, for (7) with $y = x$, $y_1(2) (= x_1(2))$ is adjacent to $u(1)$ and $u(0)$ in the Boolean network, which implies $y_1(2)$ is arbitrarily given by $u(1)$ and $u(0)$. In a similar way, $y_2(2) (= x_2(2))$ is adjacent to $u(1)$. However, $(y_1(2), y_2(2)) = (1, 1)$ cannot be realized by $u(0)$ and $u(1)$ because $y_1(2) = 0$ always

holds when $y_2(2) = 1$. These examples are very important in discussing the controllability in this paper, that is, if the Boolean function of $y_i(T)$ includes an initial state $x(0)$, or includes the same input in the outputs at the same time, then the system in question is not T -output-controllable. In the following section, by motivating the above discussion, we will consider to derive a controllability condition.

IV. OUTPUT-CONTROLLABILITY CONDITION

In this section, by using the form of an algorithm, a sufficient condition for the system (3) to be T -output-controllable will be proposed.

A. Preliminaries

In Boolean functions, identical equations are in general given by

$$h(a) \wedge (-h(a)) \equiv 0, \quad h(a) \vee (-h(a)) \equiv 1 \quad (8)$$

where $h(\cdot)$ is any Boolean function of a vector of binary variables. Obviously such identities on x_i or u_i affect the controllability in a Boolean network (note that even if $y(T) = x(0) \vee (-x(0)) \vee u(0)$, $y(T) \equiv 1$ holds irrespective of $u(0)$).

Example 1: Consider a simple example given by

$$\begin{cases} x_1(k+1) = u(k), \\ x_2(k+1) = x_1(k) \wedge (-u(k)), \\ x_3(k+1) = x_1(k) \wedge x_2(k), \\ y_1(k) = x_2(k), \\ y_2(k) = x_3(k). \end{cases} \quad (9)$$

This system has the following relation:

$$y_2(2) = x_1(0) \wedge \{u(0) \wedge (-u(0))\} = 0. \quad (10)$$

Similarly, we see that $y_2(T) = 0$, $T \geq 2$ hold identically.

In this paper, we will focus on finding such identities in $y(T)$. Before discussing a kind of initial condition and a kind of input-independency, several symbols are introduced in this subsection.

First, the following assumptions are made.

Assumption 1: The Boolean function f in (3) has no redundant variables.

For example, in the logical function $h(a, b) = a \wedge (b \vee \neg b)$, $h(a, 0) = h(a, 1)$ holds. So b is a redundant variable, and $h(a, b)$ can be rewritten as $h(a) = a$. Assumption 1 holds for any Boolean function by rewriting the system in terms of De Morgan's laws (e.g., $\neg(a \vee b) = (\neg a) \wedge (\neg b)$) and eliminating redundant variables from a given Boolean function. By Assumption 1 it is guaranteed that the Boolean function f itself does not include any identities, although $y(T)$ may include some identities. Suppose that the number of the logical NOT appeared in (3) is given by p , where the logical NOT operators are distinguished when the corresponding terms are different even if the corresponding variables are the same. In addition, consider the fictitious inputs $v_i(k) = 1$, $i = 1, 2, \dots, p$, which have one-to-one correspondence with the variables operated by the logical NOT, i.e., $\neg x_i$ or $\neg u_i$

in (3). Then the system (3) can be equivalently rewritten as the following system:

$$\Sigma_v \begin{cases} x(k+1) = f_v(x(k), u(k), v(k)), \\ y(k) = Cx(k) \end{cases} \quad (11)$$

where the Boolean function f_v does not include the logical NOT, and $v(k) = [v_1(k) \ \dots \ v_p(k)]^\top = [1 \ \dots \ 1]^\top$. For example, the system (9) is rewritten as

$$\begin{cases} x_1(k+1) = u(k), \\ x_2(k+1) = x_1(k) \wedge (v(k) \oplus u(k)), \\ x_3(k+1) = x_1(k) \wedge x_2(k), \\ y_1(k) = x_2(k), \quad y_2(k) = x_3(k) \end{cases} \quad (12)$$

subject to $v(k) = 1$.

Next, consider the adjacency matrix $\Phi \in \{0, 1\}^{(n+m+p) \times (n+m+p)}$ for the directed graph induced by the Boolean network of the system (11). For example, the adjacency matrix for the system (12) is given by

$$\Phi = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} \begin{matrix} x_1 \\ x_2 \\ x_3 \\ u \\ v \end{matrix} \quad (13)$$

$x_1 \quad x_2 \quad x_3 \quad u \quad v$

where if there exists an arc from node i to node j , then the (i, j) -th element of Φ is 1. Hereafter, without loss of generality, the following assumption is made.

Assumption 2: The i -th element of $[x^\top \ u^\top \ v^\top]^\top$ is assigned to node i in the directed graph, where $i \in \{1, 2, \dots, n+m+p\}$.

By Assumption 2, node i is identified with the i -th element of $[x^\top \ u^\top \ v^\top]^\top$. Then in Fig. 2, which shows a temporal/spatial network of the system (9), we say that for example, there exists a path between $x_2(2)$ and $u(0)$.

Using the adjacency matrix Φ , we also compute the matrix $\Phi^t C_0^\top C^\top$, $t = 1, 2, \dots, T$, where $C_0 = [I_n \ 0_{n \times (m+p)}] \in \{0, 1\}^{n \times (n+m+p)}$. For the system (11), $\Phi^t C_0^\top C^\top$ expresses whether there exist paths between $y(T)$ and $x(T-t)$, $y(T)$ and $u(T-t)$, or $y(T)$ and $v(T-t)$ for any given T . In other words, $\Phi^t C_0^\top C^\top$ expresses which elements of $x(T-t)$, $u(T-t)$ and $v(T-t)$ are variables of a Boolean function representing $y_i(T)$. However, note here that from $\Phi^t C_0^\top C^\top$, we cannot specify an explicit form of the Boolean function in question.

Furthermore, the following symbol is used: $[(X^t)^\top \ (U^t)^\top \ (V^t)^\top]^\top = \Phi^t C_0^\top C^\top$, where $X^t \in \mathcal{N}^{n \times r}$, $U^t \in \mathcal{N}^{m \times r}$, $V^t \in \mathcal{N}^{p \times r}$. Let also X_{i_x, j_x}^t , U_{i_u, j_u}^t , V_{i_v, j_v}^t denote each element of X^t , U^t , V^t , respectively. If $X_{i_x, j_x}^t \geq 1$ holds, then there exist X_{i_x, j_x}^t paths between $y_{j_x}(T)$ and $x_{i_x}(T-t)$. For the state x_{i_x} , $i_x = 1, 2, \dots, n$ of the system (3), let \mathcal{P}_x express the index set of elements of x_{i_x} operated by the logical NOT as $\neg x_{i_x}$. In a similar way, for the control input u_{i_u} , $i_u = 1, 2, \dots, m$ of the system (3), let \mathcal{P}_u express the index set of elements of u_{i_u} operated by the logical NOT as $\neg u_{i_u}$. Here, $p = |\mathcal{P}_x| + |\mathcal{P}_u|$ holds. In addition, there is a one-to-one correspondence between each

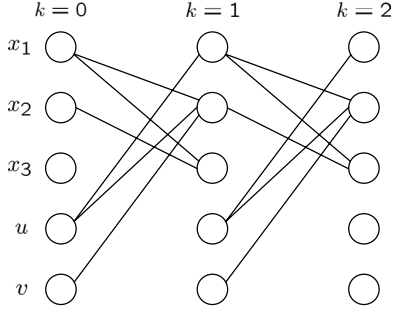


Fig. 2. Temporal/spatial network of the system (9)

element of \mathcal{P}_x , \mathcal{P}_u and the index i_v of v . Let $\nu(i_x)$ and $\nu(i_u)$ express the index i_v of v corresponding to $i_x \in \mathcal{P}_x$ and $i_u \in \mathcal{P}_u$, respectively. In the case of the system (12), $\mathcal{P}_x = \emptyset$, $\mathcal{P}_u = \{1\}$ hold, and for $i_u = 1$, $\nu(i_u) = 1$ holds.

Finally, we define the following matrices: $X_0 = C_0 \Phi^T C_0^T C^T \in \mathcal{N}^{n \times r}$, $U = \bar{B} \Phi C_0^T C^T \in \mathcal{N}^{mT \times r}$, where $\bar{B} = \text{diag}(B, B, \dots, B) \in \{0, 1\}^{mT \times (n+m+p)T}$, $B = \begin{bmatrix} 0_{m \times n} & I_m & 0_{m \times p} \end{bmatrix} \in \{0, 1\}^{m \times (n+m+p)}$, $\bar{\Phi} = \begin{bmatrix} \Phi^T & (\Phi^2)^T & \dots & (\Phi^T)^T \end{bmatrix}^T \in \mathcal{N}^{(n+m+p)T \times (n+m+p)}$.

Remark 2. Identities may not be appeared in the biological relevance. The reasons why identities are appeared are that the state is binarized and that a time-delay of the state is ignored. To overcome the latter point, a temporal Boolean network model $\xi(k+1) = f_a(\xi(k), \xi(k-1), \dots, \xi(k-T))$ has been proposed in [14]. The proposed method can also be applied to a temporal Boolean network model.

B. Proposed Algorithm

Now we propose a T -output-controllability test algorithm. Since the controllability problem is NP-hard [4], we pay our attention on deriving a sufficient condition for the controllability. Although this sufficient condition is given in the form of an algorithm, it is somewhat complex. Thus before describing an algorithm, we describe the outline of the algorithm.

First, we consider a necessary condition for $y(T)$ to include identical equations. From Fig. 2 of the example (12), we see that $y_2(2) (= x_3(2))$ in (10), which has no identities, has two paths from $u(0)$ and that $v(0)$ is connected to some node on the paths. In this way, if some identical equation exists in $y_j(T)$, there always exist more than 2 paths from $y_j(T)$ to some state and also the logical-NOT operations exist on the paths, which is a necessary condition and not necessarily a sufficient condition. Since it will spend huge time to rigorously specify the existence of identities for a large network, we consider here to exclude the cases satisfying the above necessary condition, i.e., we do not determine here the controllability in such cases.

Next, for the system that includes no identical equations, we use a kind of input-independency to determine the controllability. For example, consider the case that neither identity on u nor x exists in $y(T)$ and that $y(T)$ is expressed by $y_1(T) = h_1(u_1(0), u_2(3))$, $y_2(T) = h_2(u_1(1), u_2(1), u_2(2))$ as a result of recursive calculation (see Section 6 for such

an example), where h_1, h_2 are some Boolean functions. This system is obviously T -output controllable because each $y_j(T)$ is expressed by different $u_i(k)$ and no x_0 exists in $y_j(T)$. From the viewpoint of adjacency relation, this implies that there exists no path between $x(0)$ and $y(T)$, there exists at least one path from each $y_j(T)$ to some $u_i(k)$, and each $u_i(k)$ has a path with only one $y_j(T)$ or has no path to any $y_j(T)$. This can be easily found from the adjacency matrix, although it is a sufficient condition for the controllability. This is a rough story of our approach.

The proposed algorithm is given as follows.

T -output-controllability test algorithm:

Part A: Check of the existence of identical equations

Step 1: Set $t = 1$. Compute X^1, U^1 and V^1 .

Step 2: If $T = 1$, go to Step 6. Otherwise set $t = t + 1$. Compute X^t, U^t and V^t .

Step 3: If there exists (i_x, j_x) such that $X_{i_x, j_x}^t \geq 2$ or (i_u, j_u) such that $U_{i_u, j_u}^t \geq 2$, denote them by (i_x^*, j_x^*) or (i_u^*, j_u^*) , respectively, and go to Step 4. Otherwise, go to Step 2 if $t < T$ and go to Step 6 if $t = T$.

Step 4: If there exists i_x^* such that $i_x^* \in \mathcal{P}_x$ or i_u^* such that $i_u^* \in \mathcal{P}_u$, and $V_{\nu(i_x^*), j_x^*}^t \geq 1$ or $V_{\nu(i_u^*), j_u^*}^t \geq 1$ holds, go to Step 8. Otherwise, go to Step 5.

Step 5:

Step 5-1: Set $j = 1$.

Step 5-2: If any element of j_x^* -th column or j_u^* -th column in V^j is greater than or equal to 1, go to Step 8. Otherwise, go to Step 5-3.

Step 5-3: If $j \leq t - 1$, set $j = j + 1$ and go to Step 5-2, or else go to Step 2.

Part B: Check of the independence of each $y(T)$

Step 6: If the following conditions hold for X_0 and U , the system (3) is T -output-controllable, or else if only condition (i) does not hold, then go to Step 7. Otherwise go to Step 8.

- (i) $X_0 = 0_{n \times r}$ holds,
- (ii) Each column vector of U is a non-zero vector,
- (iii) Each row vector of U is a non-zero vector, or has only one element with a non-zero value.

Step 7: Suppose $x(0) = x_0$ for a given constant vector $x_0 \in \{0, 1\}^n$. Let $\mathcal{L} \subseteq \{1, 2, \dots, n\}(x_0)$ denote the index set of elements of $x(1) = f_v(x_0, u(0), v(0))$ that are constant for any $u(0)$ ($v(0) = 1$). Then if the following condition holds, the system (3) is T -output-controllable at x_0 . Otherwise, go to Step 8.

- (iv) For $X^{T-1} (= C_0 \Phi^{T-1} C_0^T C^T)$, there exists no $l \in \mathcal{L}(x_0)$ satisfying $X_{l, j_x}^{T-1} \geq 1$.

Step 8: This algorithm cannot determine whether the system (3) is T -output-controllable or not (at x_0).

Then the following theorem is obtained.

Theorem 1: For a given T , the following statements hold:

- (i) The system (3) is T -output-controllable if conditions

- (i), (ii) and (iii) in Step 6 hold subject to Part A.
(ii) For a given $x_0 \in \{0, 1\}^n$, the system (3) is T -output-controllable at x_0 if condition (iv) in Step 7 holds subject to Part A and Step 6.

See Appendix I for the proof. Furthermore, the computational complexity of the proposed algorithm is given by the following lemma.

Lemma 1: The computational complexity of the proposed algorithm is $O((n+m+p)^3(T-1) + (n+m+p)nrT)$ for $T \geq 2$, $n, m, p, r \geq 1$.

The proof will be omitted due to the limited space. From Lemma 1, we see that the proposed algorithm is a polynomial-time algorithm.

As an example, consider the system (9) again. Suppose $T = 2$. In Step 3, from $U_{1,2}^2 = 2$, we obtain $(i_u^*, j_u^*) = (1, 2)$ and $\nu(i_u^*) = 1$. In Step 4, from $\mathcal{P}_x = \emptyset$ and $\mathcal{P}_u = \{1\}$, we have $i_u^* \in \mathcal{P}_u$ and $V_{1,2}^2 = 1$. So go to Step 8, i.e., it is impossible to determine if the system (9) is 2-output-controllable. In fact, from (10), $y_2(2)$ includes the identity $u(0) \wedge (-u(0)) = 0$. Thus we see that there exists an identical equation. Let us also consider the case of $y = x_2$, $C = [0 \ 1 \ 0]$ in the system (9). Then for $T = 2$, we have $\Phi C_0^\top C^\top = [1 \ 0 \ 0 \mid 1 \mid 1]^\top$, $\Phi^2 C_0^\top C^\top = [0 \ 0 \ 0 \mid 1 \mid 0]^\top$. From Step 1 \rightarrow Step 2 \rightarrow Step 3 \rightarrow Step 6, we can see that the system (9) is 2-output-controllable.

As for identical equations, the proposed algorithm excludes the case of $\neg h(a) \wedge \neg h(a)$ as well as (8). This is a weak point. Furthermore, consider the system $x(k+1) = x(k) \oplus u(k)$. This system is T -output-controllable for $T = 1$. However, the proposed algorithm cannot determine whether this system is 1-output-controllable or not; thus there exists a class of systems such that the proposed algorithm cannot determine the controllability.

While the proposed algorithm includes such disadvantages, one of the main advantages of the algorithm is that the computational complexity of the above algorithm is small.

V. NUMERICAL EXAMPLE

As a numerical example, we consider the Boolean network model of a neurotransmitter signaling pathway in Fig. 3, which has been proposed in [8]. This model expresses the molecular pathway between two neurotransmitter systems, the dopamine and glutamate receptors. The state equation of this system is omitted due to the limited space (see [8]). From Fig. 3, we see that this Boolean network includes at least four loops, e.g., the loop of ξ_2, ξ_4, ξ_6 , and ξ_5 , the loop of $\xi_{11}, \xi_{12}, \xi_{14}, \xi_9$ and ξ_{10} , etc.

We consider here the case in which for a fixed dimension of $u(k)$ and the fixed output $y(k) = [\xi_8(k) \ \xi_{16}(k)]^\top$, all combinations of $\xi_i(k)$, $i = 1, 2, \dots, 7, 9, \dots, 15$ are considered as the control inputs, which we call the input-combinations, and then for a given T , the proposed algorithm is applied to the system of the form (3) obtained for each input-combination of $\xi_i(k)$. It is remarked that depending on the choice of the control inputs, there exist several cases to which the polynomial-time algorithm proposed in [4] cannot be applied due to the graph-structure constraints.

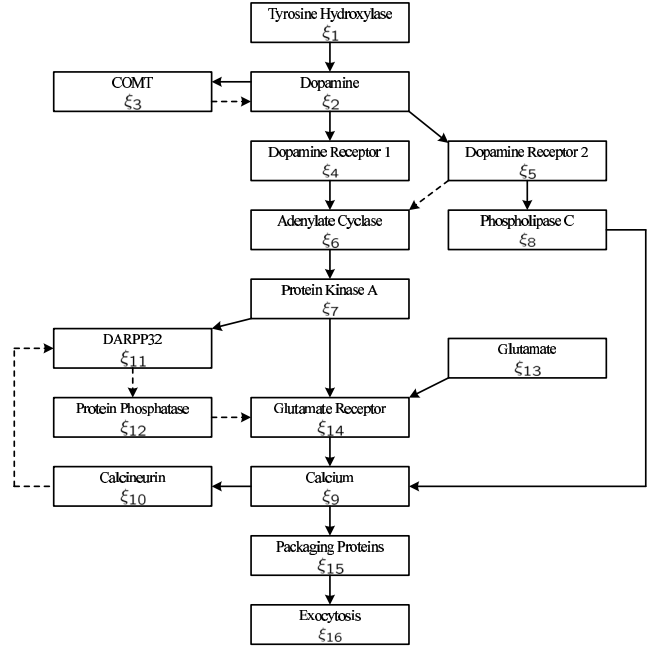


Fig. 3. Simplified model of interaction pathway between the glutamatergic and dopaminergic receptors. Activation (solid), Inhibition (broken).

By applying our algorithm to the case of each input-combination of ξ_i and each fixed T , we can obtain, for example, the following results. In the case of $\dim u(k) = 2$ and $T = 5$, there exist at least 6 input-combinations of ξ_i that makes the system 5-output-controllable.

On the other hand, in the case of $\dim u(k) = 4$ and $T = 6$, we obtain as one of combinations of $x(k) \in \{0, 1\}^{12}$ and $u(k) \in \{0, 1\}^4$ that make the system 6-output-controllable

$$x(k) = [\xi_1(k) \ \xi_3(k) \ \xi_4(k) \ \xi_5(k) \ \xi_6(k) \ \xi_8(k) \ \xi_9(k) \ \xi_{11}(k) \ \xi_{12}(k) \ \xi_{14}(k) \ \xi_{15}(k) \ \xi_{16}(k)]^\top, \quad (14)$$

$$u(k) = [\xi_2(k) \ \xi_7(k) \ \xi_{10}(k) \ \xi_{13}(k)]^\top. \quad (15)$$

In fact, based on the above result we can verify that $y(6)$ is given as $y_1(6) = u_1(4)$, $y_2(6) = u_1(1) \vee (u_2(2) \wedge \neg u_3(0) \wedge u_4(2))$, which implies that $y(6)$ can be freely given by control inputs. It is remarked that the polynomial-time algorithm proposed in [4] cannot be applied to the system with the state (14) and the input (15) because the network includes the two loops, i.e., the loop of ξ_2, ξ_4, ξ_6 , and ξ_5 , the loop of $\xi_7, \xi_{11}, \xi_{12}$, and ξ_{14} . It is expected that such a result will be one of guidelines in experimental approaches to the control problem of biological systems.

VI. CONCLUSION

In this paper, a sufficient condition for a Boolean network model with inputs and outputs to be output-controllable has been derived by exploiting an adjacency matrix of its network graph. The obtained condition, which is given in the form of an algorithm, can be checked in polynomial time with respect to the state/input dimensions and the control time; thus it will be one of the powerful tools that can provide

some clues for finding effective control inputs to control a large-scale genetic network.

There are many interesting open problems to be addressed in the future. For example, it is one of the significant topics to apply the proposed approach to a practical and large-scale gene network with more than 100 dimensions, and to show the efficiency of our approach from the experimental point of view.

ACKNOWLEDGMENT

The authors would like to thank Prof. Tatsuya Akutsu, Kyoto University for fruitful discussions and valuable comments.

APPENDIX I PROOF OF THEOREM 1

In this appendix, Theorem 1 is proven.

Proof: Noting that the identical equations have the form in (8), and $x(T)$ is obtained recursively from (11), the identical equations appeared in $x(T)$ always have the form

$$(V_1 \oplus w(k)) \wedge (V_2 \oplus w(k)) (\equiv 0), \quad (16)$$

$$(V_1 \oplus w(k)) \vee (V_2 \oplus w(k)) (\equiv 1) \quad (17)$$

where $w(k)$ denotes either variable of $x(k)$ or $u(k)$,

$$V_1 = \bigoplus_{(i,j) \in \mathcal{I}_1} v_i(k+j), \quad V_2 = \bigoplus_{(i,j) \in \mathcal{I}_2} v_i(k+j),$$

$$V_1 \oplus w(k) = w(k), \quad V_2 \oplus w(k) = \neg w(k), \quad v_i = 1$$

and $\mathcal{I}_1, \mathcal{I}_2$ are some subsets of the index set $\{(i, j) \mid i = 1, 2, \dots, p; j = 0, 1, \dots, T-1\}$.

Thus the first story of the proof is to exclude the case that $y(T)$ includes the identities that have the form of (16) or (17), from the viewpoint of a necessary condition for the identity to exist in $y(T)$.

In Step 3, from $X_{i_x, j_x}^t \geq 2$ for some t , $i_x = i_x^*$ and $j_x = j_x^*$, we see that more than 2 paths from $y_{j_x}(T)$ to $x_{i_x}(T-t)$ exist, which is necessary for the identity on $x_{i_x}(T-t)$ to exist (Similarly for the case of $U_{i_u, j_u}^t \geq 2$). Thus we next focus on the existence of logical NOT (i.e., v_i) in these paths in Step 4 and Step 5.

Consider the case that the logical NOT (i.e., v_i) corresponding to $x_{i_x^*}(T-t)$ or $u_{i_u^*}(T-t)$ obtained in Step 3 exists in (11), in other words, either $i_x^* \in \mathcal{P}_x$ or $i_u^* \in \mathcal{P}_u$ holds. Then the condition $V_{\nu(i_x^*), j_x^*}^t \geq 1$ implies that the term $v_{\nu(i_x^*)}(T-t) \oplus x_{i_x^*}(T-t)$ is included in the paths in question, which is a necessary condition for the existence of the identity in $y_{j_x^*}(T)$. Thus we exclude this case (Step 4). (Similarly for the case $u_{i_u^*}(T-t)$).

In the other case, from (16), (17), for $v(T-j)$, some $j \in \{1, 2, \dots, t-1\}$, to exist in the paths in question is necessary for the existence of identities. If any element of the j_x^* -column or the j_u^* -column of V^j is greater than or equal to 1, some element of $v(T-j)$ exists in the paths in question. Thus we exclude this case (Step 5-2).

The next story is to show that for the system in the case that $y(T)$ includes neither identities of (16) nor identities of (17), it is T -output-controllable (at x_0).

First, the proof of the statement (i) is given. Condition (i) in Step 6 implies that there exists no path between each element of $x(0)$ and each element of $y(T)$, since the (i, j) -th element of X_0 expresses if a path from $x_i(0)$ to $y_j(T)$ exists or not. On the other hand, note that $(mh+i, j)$ -th element of U expresses if a path from $u_i(T-h-1)$ to $y_j(T)$ exists or not ($h = 0, 1, \dots, T-1$). Thus condition (ii) in Step 6 implies that there exists at least one path from each element of $y(T)$ to some $u_i(k)$.

Furthermore, condition (iii) in Step 6 means that the input $u_i(k)$ for each $i \in \{1, 2, \dots, m\}$ and $k \in \{0, 1, \dots, T-1\}$ has a path connected to only one element of $y(T)$, or has no path to any element of $y(T)$. From these conditions, it follows that each $u_i(k)$ affects at most one $y_j(T)$ and not the other $y_h(T)$, $h \neq j$. Hence the value of $y_j(T)$ can be independently specified by the corresponding $u_i(k)$, which implies that the system (11) is T -output-controllable.

Next, the statement (ii) is proven. Since condition (i) in Step 6 does not hold, in this case, there exists a path between some element of $x(0)$ and some element of $y(T)$. On the other hand, condition (iv) in Step 7 guarantees that there exists no path between constant elements of $x(1) = f_v(x_0, u(0), v(0))$ and elements of $y(T)$. Thus $y(T)$ is not affected by the value of x_0 . Therefore, from (ii)-(iv), it follows that the system (11) is T -output-controllable at x_0 . This completes the proof. ■

REFERENCES

- [1] Joint Special Issue on Systems Biology, *IEEE Trans. on Automatic Control & IEEE Trans. on Circuits and Systems I*, January, 2008.
- [2] M. Aldana: Boolean dynamics of networks with scale-free topology, *Physica D*, vol. 185, pp. 45–66, 2003.
- [3] S. Azuma, E. Yanagisawa, and J. Imura: Controllability Analysis of Biosystems Based on Piecewise Affine Systems Approach, *IEEE Trans. on Automatic Control*, vol. 53, no. 1, pp. 139–152, 2008.
- [4] T. Akutsu, M. Hayashida, W.-K. Ching, and M. K. Ng: Control of Boolean networks: Hardness results and algorithms for tree structured networks, *Journal of Theoretical Biology*, vol. 244, pp. 670–679, 2007.
- [5] C. Belta, J. Schug, T. Dang, V. Kuma, G. J. Pappas, H. Rubin, and P. Dunlap: Stability and reachability analysis of a hybrid model of luminescence in the marine bacterium *Vibrio fischeri*, *Proc. of the 40th IEEE Conf. on Decision and Control*, pp. 869–874, 2001.
- [6] A. Fauré, A. Naldi, C. Chaouiya, and D. Thieffry: Dynamical analysis of a generic Boolean model for the control of the mammalian cell cycle, *Bioinformatics*, vol. 22, no. 14, pp. 124–131, 2006.
- [7] G. Ferrari-Trecate and J. Lygeros Eds., *Proc. 45th IEEE Conf. on Decision and Control/Workshop Hybrid Systems Biology*, 2006.
- [8] S. Gupta, S. S. Bisht, R. Kukreti, S. Jain, and S. K. Brahmachari: Boolean network analysis of a neurotransmitter signaling pathway, *Journal of Theoretical Biology*, vol. 244, pp. 469–469, 2007.
- [9] H. D. Jong: Modeling and Simulation of Genetic Regulatory Systems: A Literature Review, *Journal of Computational Biology*, vol. 9, no. 1, pp. 67–103, 2002.
- [10] S. Kauffman: Metabolic stability and epigenesis in randomly constructed genetic nets, *Journal of Theoretical Biology*, vol. 22, pp. 437–467, 1969.
- [11] S. Kauffman: *The Origins of Order: Self-Organization and Selection in Evolution*, Oxford University Press, 1993.
- [12] S. Martin, Z. Zhang, A. Martino, and J.-L. Faulon: Boolean dynamics of genetic regulatory networks inferred from microarray time series data, *Bioinformatics*, vol. 23, no. 7, pp. 866–874, 2007.
- [13] I. Shmulevich and W. Zhang: Binary analysis and optimization-based normalization of gene expression data, *Bioinformatics*, vol. 18, no. 4, pp. 555–565, 2002.
- [14] A. Silvescu and V. Honavar: Temporal Boolean Network Models of Genetic Networks and Their Inference from Gene Expression Time Series, *Complex Systems*, vol. 13, 54–71, 2000.