SCIENCE CHINA
Information Sciences

• **RESEARCH PAPER** •
Special Focus on Biomolecular Network Analysis and Application

July 2016, Vol. 59 070107:1–070107:14
doi: 10.1007/s11432-016-5579-8

Pinning controllability of autonomous Boolean control networks

Hongwei CHEN[1], Jinling LIANG[1*] & Zidong WANG[2,3]

[1]*Department of Mathematics, Southeast University, Nanjing* 210096*, China;*
[2]*Department of Computer Science, Brunel University London, Uxbridge, Middlesex* UB8 3PH*, UK;*
[3]*Faculty of Engineering, King Abdulaziz University, Jeddah* 21589*, Saudi Arabia*

Received April 10, 2016; accepted April 18, 2016; published online June 7, 2016

Abstract Autonomous Boolean networks (ABNs), which are developed to model the Boolean networks (BNs) with regulatory delays, are well known for their advantages of characterizing the intrinsic evolution rules of biological systems such as the gene regulatory networks. As a special type of ABNs with binary inputs, the autonomous Boolean control networks (ABCNs) are introduced for designing and analyzing the therapeutic intervention strategies where the binary inputs represent whether a certain medicine is dominated or not. An important problem in the therapeutic intervention is to design a control sequence steering an ABCN from an undesirable location (implying a diseased state) to a desirable one (corresponding to a healthy state). Motivated by such background, this paper aims to investigate the reachability and controllability of ABCNs with pinning controllers. Several necessary and sufficient criteria are provided by resorting to the semi-tensor product techniques of matrices. Moreover, an effective pinning control algorithm is presented for steering an ABCN from any given states to the desired state in the shortest time period. Numerical examples are also presented to demonstrate the results obtained.

Keywords autonomous Boolean networks, semi-tensor product, controllability, pinning control scheme, network transition matrix

Citation Chen H W, Liang J L, Wang Z D. Pinning controllability of autonomous Boolean control networks. Sci China Inf Sci, 2016, 59(7): 070107, doi: 10.1007/s11432-016-5579-8

1 Introduction

From a translational perspective, the ultimate objective of genomic research is to uncover the mechanisms with which cells execute and control the enormous number of operations required for normal functions and the ways in which cellular systems fail in disease. A rather wide spectrum of approaches have been developed to model the genetic regulatory networks (GRNs), and the most frequently investigated models include the Boolean model, the Bayesian network model and the differential equations model [1,2]. Among these models, the Boolean network (BN), originally proposed by Kauffman in 1969 [3], has been proven to be a prominent qualitative tool for modeling the genetic regulatory process. In the past few decades, the study of BNs has received considerable research attention from both the biology and the

* Corresponding author (email: jinlliang@seu.edu.cn)

physics communities. Many excellent results have been available in the literature including the topological structure [4] and the controllability of BNs [5].

In a BN, each gene is approximated as a Boolean node that switches between ON and OFF (1 and 0, respectively), and the state of a node is updated at discrete time instants according to a pre-assigned logic function which depends on the states of that node's inputs on the previous time instant. As pointed out in [6], many biological systems have exogenous perturbations that can be described as 'controls', and the concept of Boolean control networks (BCNs) has been put forward by adding binary inputs. For instance, when modeling the progression of a disease by BCNs, the binary input may represent whether a certain medicine is administered or not at each time instant.

Actually, a BCN can be regarded as a family of BNs in which the value of the control determines which BN is active. In this setup, the interest in the control problem for BCNs corresponding to therapeutic interventions arises primarily in the field of systems biology. However, due to the lack of effective tools to deal with logical systems, no unified criterion has been available for testing the controllability until the introduction of the semi-tensor product (STP) method originally proposed in [7]. Thanks to this novel STP technique, the logical dynamics of a BN (BCN) can be uniquely transformed into a standard discrete-time linear (bilinear) dynamical system. Consequently, several analysis and control problems, which include but are not limited to, controllability and observability [8–10], stability and stabilization [11–14], optimal control [15, 16], system decomposition [17, 18], have been extensively investigated in recent years. For more details about the STP, we refer the readers to [19, 20].

On the other hand, time delays in genetic regulatory process are inevitable primarily due to the slow processes of transcription, translation and translocation or the finite switching speed of amplifiers. It has now been well recognized that time delays in GRNs may play an important role in the predictions of the dynamics of mRNA and protein concentrations, and theoretical models without consideration of these delay factors may even have led to wrong predictions [21]. As such, when modeling the dynamic behaviors of GRNs by BCNs, the inherent time delays should be taken into account. Note that the controllability issue for BCNs with time delays has emerged as a research topic of great importance. In [22], the controllability of BCNs with time-invariant delays in states has been studied by increasing the dimension of the state space. This approach has been further adopted in [23] and [24] to deal with the controllability of higher-order BCNs. In [25], the authors have considered the controllability of time-variant BCNs as well as BCNs with multiple time-variant bounded state delays. An equivalent test criterion for the controllability of BCNs with unbounded time-delays in states has been given in [26] based on a new proposed concept called controllability constructed path.

A recent yet significant discovery in the cellular reprogramming field is that full control and reprogramming of biological systems may be achieved by controlling only a few key factors [27]. This discovery seems to be in contradiction to the conventional definition of controllability of BCNs that concerns with the control of all the system's nodes. As a matter of fact, for a large-scale GRN, it is usually difficult to add controllers to all nodes. In order to reduce the number of controller, a natural idea is to control the network by pining only part of the nodes. In [28], a BN model has been developed to reproduce the two-phase dynamics of the p53 network in response to DNA damage, where a practical control scheme has been proposed by pinning the state of a critical node to steer the system to the desired attractor pertaining to the desired final state. It is worth emphasizing that such kind of pinning control scheme is of paramount importance for medical treatment and genetic engineering. Recently, by resorting to the STP technique, the pinning controllability of BCNs has been investigated in [29] and the pinning control design for stabilizing the BNs has been given in [30].

In [31], Ghil and Mullhaupt have introduced the Boolean delay equations as an autonomous BN (ABN), and this model has then been widely applied to yeast cell cycle and electronic circuits [32]. In [33], based on the logic gates, Rivera-Durón has built an electronic realization of an ABN of five nodes with external Boolean signal that can be regarded as control input, namely autonomous Boolean control network (ABCN) (see Example 3 for reference), and the forced synchronization issue of two identical ABNs (forced by a common external signal) has been addressed. The recent work in [34] has shown that the analysis on complex dynamics in an electronic circuit is beneficial for capturing the qualitative aspects of the structure and dynamics of GRNs. Very recently, Cheng et al. [35] have first presented the ABN framework in mathematical terms, described the biological meaning of its time-delay parameters, and then applied it to the *Drosophila* segment polarity gene regulatory system. Experimental results have further confirmed that important timing information associated with the regulatory interactions among genes can be faithfully represented in ABN models, and that such models can provide a direct insight into understanding and controlling the GRNs. All the above theoretical and experimental results indicate that the ABCN is an important and appropriate model to simulate and analyze the GRNs.

In this paper, by following the main stream of research, we further consider the pining controllability problem of ABCNs in which only a selected fraction of nodes (not every node of the ABCNs) are controlled. Based on the algebraic representation of logical dynamics in terms of STP of matrices, the inherent special structures of the network transition matrix are investigated, and matrix testing criteria for the pinning controllability of ABCNs are obtained. Then, we further devise practical control schemes for steering an ABCN between two given states in a given number of time-steps by pinning a selected fraction of nodes. The approach proposed offers insights into understanding and controlling the practical biological systems, which is of paramount importance for the therapeutic intervention and in the genetic engineering.

The rest of the paper is organized as follows. Section 2 contains some notations and preliminaries on the STP of matrices. The main results of this paper are presented in Section 3, and a brief conclusion is drawn in Section 4.

2 Preliminaries

The following notations will be used throughout this paper.

- $\mathcal{Z}$ and $\mathcal{N}$ are the sets of integers and nonnegative integers, respectively.
- $\mathcal{D} := \{1, 0\}$, and $\mathcal{D}^n = \underbrace{\mathcal{D} \times \cdots \times \mathcal{D}}_{n}$.
- $\mathbb{R}^{m\times n}$ means the set of all $m \times n$ real matrices.
- $\Delta_m := \{\delta_m^1, \delta_m^2, \dots, \delta_m^m\}$, where δ_m^i is the ith column of the identity matrix I_m for $i = 1, 2, \dots, m$.
- A matrix $A \in \mathbb{R}^{m\times n}$ is called a logical matrix if $A = [\delta_m^{i_1}\ \delta_m^{i_2}\ \cdots\ \delta_m^{i_n}]$ $(i_1, i_2, \dots, i_n \in \{1, 2, \dots, m\})$, which is also expressed by $A = \delta_m[i_1, i_2, \dots, i_n]$ for simplicity. The set of all $m \times n$ logical matrices is denoted by $\mathcal{L}_{m\times n}$.
- $\mathrm{Col}_i(A)$ (respectively, $\mathrm{Row}_i(A)$) is used to represent the ith column (respectively, row) of matrix A.
- $\mathrm{Blk}_i(A)$ represents the ith block of matrix $A = [A_1\ A_2\ \cdots\ A_p]$, where all A_i $(i = 1, 2, \dots, p)$ have the same dimensions.
- $\mathbf{1}_m$ denotes the m-dimensional column vector with all entries being 1, i.e., $\mathbf{1}_m = \sum_{k=1}^m \delta_m^k$, and $\mathbf{1}_m^{\mathrm{T}} = [1\ 1\ \cdots\ 1]$, where 'T' represents the transpose of $\mathbf{1}_m$.
- For $A \in \mathbb{R}^{m\times n}$ and $B \in \mathbb{R}^{r\times n}$, the Khatri-Rao product of A and B, denoted by $A * B$, is defined as $A * B = [\mathrm{Col}_1(A) \otimes \mathrm{Col}_1(B)\ \cdots\ \mathrm{Col}_n(A) \otimes \mathrm{Col}_n(B)]$, where '$\otimes$' is the Kronecker product.

Firstly, the definition and some basic properties of STP are introduced that are useful in our later discussion.

Definition 1 ([7]). The STP of two matrices $A \in \mathbb{R}^{m\times n}$ and $B \in \mathbb{R}^{p\times q}$ is defined as

$$A \ltimes B = (A \otimes I_{\alpha/n})(B \otimes I_{\alpha/p}),$$

where $\alpha = \mathrm{lcm}(n, p)$ is the least common multiple of n and p.

Remark 1. When $n = p$, $A \ltimes B = (A \otimes I_1)(B \otimes I_1) = AB$. Therefore, the STP is a generalization of the conventional matrix product that provides a way to make two matrices with arbitrary dimensions multiplicable. Hereafter, we simply call it 'product' and omit the symbol '$\ltimes$' if no confusion arises.

Definition 2 ([7]). A swap matrix $W_{[m,n]}$ is an $mn \times mn$ matrix defined as follows: its rows and columns are labeled by double index (i, j), where the columns are arranged by the ordered multi-index $\mathrm{Id}(i, j; m, n)$ and the rows are arranged by the ordered multi-index $\mathrm{Id}(j, i; n, m)$. The element at the position $((I,J),(i,j))$ is then set to

$$w_{(I,J),(i,j)} = \delta_{i,j}^{I,J} = \begin{cases} 1, & I = i \text{ and } J = j, \\ 0, & \text{otherwise.} \end{cases}$$

Lemma 1 ([7])**.** The STP of matrices has the following properties:

(1) Let $X \in \Delta_m$ and $Y \in \Delta_n$, then $Y \ltimes X = W_{[m,n]} \ltimes X \ltimes Y$, where $W_{[m,n]}$ is the swap matrix.

(2) Let the $k^2 \times k$ logical matrix $\Psi_k = [\delta_k^1 \otimes \delta_k^1 \ \delta_k^2 \otimes \delta_k^2 \ \cdots \ \delta_k^k \otimes \delta_k^k]$ and $Z \in \Delta_k$, then $Z \ltimes Z = \Psi_k Z$.

Secondly, by identifying $1 \sim \delta_2^1$ and $0 \sim \delta_2^2$, where '$\sim$' means two different/equivalent forms of the same object, the logical variable in $\mathcal{D}$ then takes value from Δ_2. And then a logical function with n variables in $\mathcal{D}$ can be expressed in the algebraic form as follows.

Lemma 2 ([7])**.** Let $f(x_1, x_2, \ldots, x_n) : \mathcal{D}^n \rightarrow \mathcal{D}$ be a logical function. Then there exists a unique matrix $M_f \in \mathcal{L}_{2\times 2^n}$, called the structure matrix of f, such that

$$f(x_1, x_2, \ldots, x_n) = M_f \ltimes_{i=1}^n x_i, \quad x_i \in \Delta_2,$$

where $\ltimes_{i=1}^n x_i = x_1 \ltimes x_2 \ltimes \cdots \ltimes x_n$.

Finally, to proceed, some properties of the Kronecker product are presented which will be used in the sequel.

Lemma 3. The following results hold for the Kronecker product of matrices:

(1) If matrices A, B, C and D have proper dimensions, then

$$(A \otimes B)(C \otimes D) = (AC) \otimes (BD).$$

(2) If $X \in \Delta_m$, $Y \in \Delta_n$, then $\mathbf{1}_m^{\mathrm{T}} X = 1$, $\mathbf{1}_n^{\mathrm{T}} Y = 1$, $X \ltimes Y = X \otimes Y$ and

$$X = (I_m X) \otimes (\mathbf{1}_n^{\mathrm{T}} Y) = (I_m \otimes \mathbf{1}_n^{\mathrm{T}}) XY, \quad Y = (\mathbf{1}_m^{\mathrm{T}} X) \otimes (I_n Y) = (\mathbf{1}_m^{\mathrm{T}} \otimes I_n) XY.$$

3 Main results

3.1 Algebraic representation form of the ABCNs

A conventional BCN with m controllers and n nodes can be described by the following discrete-time logical dynamic system

$$x_i(t+1) = f_i(u_1(t), \ldots, u_m(t), x_1(t), \ldots, x_n(t)), \quad i = 1, \ldots, n, \tag{1}$$

where x_i and u_j $(j = 1, 2, \ldots, m)$ are the ith state variable and the control input variables taking values in $\mathcal{D}$, $f_i : \mathcal{D}^{m+n} \rightarrow \mathcal{D}$ $(i = 1, \ldots, n)$ are the Boolean functions, $t \in \mathcal{Z}$ satisfies $t \geqslant t_0$ with $t_0 \in \mathcal{Z}$ being the initial time.

In many cases of interest, signals propagate in the network with such a slow speed so that the time delays along the links are comparable to or larger than the characteristic response times of the nodes. In this case, time delays must be introduced in the model which can be described by a set of numbers $\{\tau_{ij} \in \mathcal{N} : i, j = 1, \ldots, n\}$, where τ_{ij} is the time for node x_j to have an effect on node x_i, i.e., the time that a signal takes to propagate to node i from node j. And the delayed feedbacks among the Boolean variables are characterized by the following ABCNs:

$$x_i(t+1) = f_i(u_1(t), \ldots, u_m(t), x_1(t-\tau_{i1}), \ldots, x_n(t-\tau_{in})), \quad i = 1, \ldots, n. \tag{2}$$

For example, in GRNs, time delays are ubiquitous due to the slow biochemical reactions in the process of transcription, translation, and degradation. And the recent publication [35] shows that the ABCN is just an important and appropriate model to simulate and analyze the dynamics of GRNs.

On the other hand, for some special cases of biological systems such as mammalian cells, full control effect might also be achieved by controlling only a few key factors [27], which seems to conflict with the case for the conventional ABCNs where all nodes are required to exert full control. This demonstrates that when investigating the biological systems researches on the pinning control of ABCNs are not only meaningful but also necessary. Based on the above discussions, in this paper, we consider the pinning controllability of ABCNs with n nodes, where the nodes $i_1, i_2, \ldots, i_r$ $(1 \leqslant r < n)$ are selected to be controled. Without loss of generality, we assume that $i_k = k$ $(k = 1, 2, \ldots, r)$. Then, the dynamics of ABCNs with r pinning controllers can be described as follows:

$$\begin{cases} x_1(t+1) = f_1(u_1(t), x_1(t-\tau_{11}), \ldots, x_n(t-\tau_{1n})), \\ \qquad \vdots \\ x_r(t+1) = f_r(u_r(t), x_1(t-\tau_{r1}), \ldots, x_n(t-\tau_{rn})), \\ x_{r+1}(t+1) = f_{r+1}(x_1(t-\tau_{r+1,1}), \ldots, x_n(t-\tau_{r+1,n})), \\ \qquad \vdots \\ x_n(t+1) = f_n(x_1(t-\tau_{n1}), \ldots, x_n(t-\tau_{nn})). \end{cases} \tag{3}$$

Let $x = \ltimes_{k=1}^{n} x_k$ and $\tau = \max_{1 \leqslant i,j \leqslant n}\{\tau_{ij}\}$, combining with Lemmas 1–3, system (3) can be transformed into the component-wise algebraic form as follows:

$$x_i(t+1) = \hat{F}_i u_i(t) x(t) \cdots x(t-\tau), \quad i = 1, \ldots, r, \tag{4a}$$

$$x_j(t+1) = \hat{F}_j x(t) \cdots x(t-\tau), \quad j = r+1, \ldots, n, \tag{4b}$$

where $\hat{F}_i \in \mathcal{L}_{2 \times 2^{n(\tau+1)+1}}$ and $\hat{F}_j \in \mathcal{L}_{2 \times 2^{n(\tau+1)}}$. By further denoting $x^1 = \ltimes_{k=1}^{r} x_k$, $x^2 = \ltimes_{k=r+1}^{n} x_k$ and $u = \ltimes_{k=1}^{r} u_k$, it follows from (4) that

$$x^1(t+1) = F_1 u(t) x(t) \cdots x(t-\tau), \tag{5a}$$

$$x^2(t+1) = F_2 x(t) \cdots x(t-\tau), \tag{5b}$$

where $F_1 = (\otimes_{k=1}^{r} \hat{F}_k) \ltimes_{k=1}^{r-1} [(I_{2^k} \otimes W_{[2, 2^{n(\tau+1)}]})(I_{2^{k+1}} \otimes \Psi_{2^{n(\tau+1)}})] \in \mathcal{L}_{2^r \times 2^{n(\tau+1)+r}}$ and $F_2 = *_{k=r+1}^{n} \hat{F}_k \in \mathcal{L}_{2^{n-r} \times 2^{n(\tau+1)}}$ in which '$*$' is the Khatri-Rao product. By (5b) and Lemma 3, one obtains

$$\begin{aligned} x^2(t+1) &= F_2(\mathbf{1}_{2^r}^{\mathrm{T}} \otimes I_{2^{n(\tau+1)}}) u(t) x(t) \cdots x(t-\tau) = (\mathbf{1}_{2^r}^{\mathrm{T}} \otimes F_2) u(t) x(t) \cdots x(t-\tau) \\ &\triangleq \bar{F}_2 u(t) x(t) \cdots x(t-\tau), \end{aligned} \tag{6}$$

where $\bar{F}_2 = (\mathbf{1}_{2^r}^{\mathrm{T}} \otimes F_2) \in \mathcal{L}_{2^{n-r} \times 2^{n(\tau+1)+r}}$. This together with (5a) gives the following algebraic form of system (3):

$$x(t+1) = F u(t) x(t) \cdots x(t-\tau), \tag{7}$$

where $F = F_1 * \bar{F}_2 \in \mathcal{L}_{2^n \times 2^{n(\tau+1)+r}}$, called the *network transition matrix* of ABCN (3).

3.2 Structure of the network transition matrix

In the following, we further investigate the inherent special structure of the network transition matrix F in the algebraic equation (7). Suppose that $F_1 = \delta_{2^r}[i_1, i_2, \ldots, i_{2^{n(\tau+1)+r}}]$ and $F_2 = \delta_{2^{n-r}}[j_1, j_2, \ldots, j_{2^{n(\tau+1)}}]$, one has $\bar{F}_2 = [\underbrace{F_2 \ \cdots \ F_2}_{2^r}]$. Note that $F = F_1 * \bar{F}_2$, it thus follows that

$$\begin{aligned} F = \delta_{2^n}[&(i_1 - 1)2^{n-r} + j_1, \ldots, (i_{2^{n(\tau+1)}} - 1)2^{n-r} + j_{2^{n(\tau+1)}}, \\ &(i_{2^{n(\tau+1)}+1} - 1)2^{n-r} + j_1, \ldots, (i_{2^{n(\tau+1)} \cdot 2} - 1)2^{n-r} + j_{2^{n(\tau+1)}}, \\ &\cdots \\ &(i_{2^{n(\tau+1)} \cdot (2^r - 1)+1} - 1)2^{n-r} + j_1, \ldots, (i_{2^{n(\tau+1)} \cdot 2^r} - 1)2^{n-r} + j_{2^{n(\tau+1)}}]. \end{aligned} \tag{8}$$

Theorem 1. Let (7) be the algebraic representation of ABCN (3). Then the network transition matrix for the ABCN (3) must be in the form of (8).

Example 1. Consider the following ABCN with three nodes and two pinning controllers:

$$\begin{cases} x_1(t+1) = u_1(t) \leftrightarrow [x_1(t) \vee x_2(t-1)], \\ x_2(t+1) = u_2(t) \vee x_3(t-2), \\ x_3(t+1) = x_1(t) \wedge x_2(t-1), \end{cases} \tag{9}$$

where '$\leftrightarrow$', '$\vee$', and '$\wedge$' represent the logical functions XNOR, OR, and AND, respectively. Denote $x^1 = \ltimes_{k=1}^{2} x_k$, $x^2 = x_3$, $x = \ltimes_{k=1}^{3} x_k$, and $u = \ltimes_{k=1}^{2} u_k$, by resorting to the STP technique, one can obtain the following algebraic representation

$$\begin{cases} x^1(t+1) = F_1 u(t)x(t)x(t-1)x(t-2), \\ x^2(t+1) = F_2 x(t)x(t-1)x(t-2), \end{cases}$$

where $F_1 = \delta_4[\underbrace{1,1,\ldots,3}_{2^9},\underbrace{1,2,\ldots,4}_{2^9},\underbrace{3,3,\ldots,1}_{2^9},\underbrace{1,3,\ldots,2}_{2^9}]$ and $F_2 = \delta_2[\underbrace{1,1,\ldots,2}_{2^9}]$. Then the network transition matrix F can be calculated as follows:

$$\begin{aligned} F = \delta_8[&(1-1)2+1,(1-1)2+1,\ldots,(3-1)2+2, \\ &(1-1)2+1,(2-1)2+1,\ldots,(4-1)2+2, \\ &(3-1)2+1,(3-1)2+1,\ldots,(1-1)2+2, \\ &(1-1)2+1,(3-1)2+1,\ldots,(2-1)2+2], \end{aligned} \tag{10}$$

which is just in the form of (8).

Now, we consider the following type of ABCN with n nodes and s nodes ($1 \leqslant s < n$) are selected to be controlled. Without loss of generality, the first s nodes are assumed to be controlled:

$$\begin{cases} x_1(t+1) = g_1(u_1(t),\ldots,u_r(t),x_1(t-\tau_{11}),\ldots,x_n(t-\tau_{1n})), \\ \quad\vdots \\ x_s(t+1) = g_s(u_1(t),\ldots,u_r(t),x_1(t-\tau_{s1}),\ldots,x_n(t-\tau_{sn})), \\ x_{s+1}(t+1) = g_{s+1}(x_1(t-\tau_{s+1,1}),\ldots,x_n(t-\tau_{s+1,n})), \\ \quad\vdots \\ x_n(t+1) = g_n(x_1(t-\tau_{n1}),\ldots,x_n(t-\tau_{nn})). \end{cases} \tag{11}$$

Similar to system (3), denoting $x^1 = \ltimes_{k=1}^{s} x_k$, $x^2 = \ltimes_{k=s+1}^{n} x_k$, $x = \ltimes_{k=1}^{n} x_k$, and $u = \ltimes_{k=1}^{r} u_k$, a direct computation using the properties of STP can easily produce the following algebraic representation of system (11):

$$x^1(t+1) = G_1 u(t)x(t)\cdots x(t-\tau), \tag{12a}$$

$$x^2(t+1) = G_2 x(t)\cdots x(t-\tau). \tag{12b}$$

Let $G_1 = \delta_{2^s}[i_1,i_2,\ldots,i_{2^{n(\tau+1)+r}}]$ and $G_2 = \delta_{2^{n-s}}[j_1,j_2,\ldots,j_{2^{n(\tau+1)}}]$, an algebraic form of system (11) can be obtained as

$$x(t+1) = Gu(t)x(t)\cdots x(t-\tau), \tag{13}$$

where the transition matrix G satisfies the following form:

$$\begin{aligned} G = \delta_{2^n}[&(i_1-1)2^{n-s}+j_1,\ldots,(i_{2^{n(\tau+1)}}-1)2^{n-s}+j_{2^{n(\tau+1)}}, \\ &(i_{2^{n(\tau+1)}+1}-1)2^{n-s}+j_1,\ldots,(i_{2^{n(\tau+1)}.2}-1)2^{n-s}+j_{2^{n(\tau+1)}}, \\ &\cdots \\ &(i_{2^{n(\tau+1)}.(2^r-1)+1}-1)2^{n-s}+j_1,\ldots,(i_{2^{n(\tau+1)}.2^r}-1)2^{n-s}+j_{2^{n(\tau+1)}}]. \end{aligned} \tag{14}$$

Corollary 1. Let (13) be the algebraic representation of ABCN (11). Then the network transition matrix of the system (11) must be in the form of (14).

Comparing Theorem 1 with Corollary 1, one can find that the network transition matrix G has the same form with that of F when $s = r$. In other words, if $s = r$, then systems (3) and (11) have the same structural algebraic representation (7), where G in (14) is in the same form of (8). Therefore, for a given network transition matrix in the form of (8), one might need to judge whether the dynamic of the logical network is expressed in the framework of (3) or (11). In the following, we will investigate this problem. To this end, we first define a set of logical matrices $S_i^r \in \mathcal{L}_{2\times 2^r}$, called retrievers, as follows

$$\begin{aligned} S_i^r &= \delta_2[\underbrace{1,\ldots,1}_{2^{r-i}},\underbrace{2,\ldots,2}_{2^{r-i}},\ldots,\underbrace{1,\ldots,1}_{2^{r-i}},\underbrace{2,\ldots,2}_{2^{r-i}}] \\ &\triangleq [\mathrm{Blk}_1(S_i^r)\ \mathrm{Blk}_2(S_i^r)\ \cdots\ \mathrm{Blk}_{2^i}(S_i^r)], \quad i = 1,\ldots,r, \end{aligned} \tag{15}$$

where $\mathrm{Blk}_p(S_i^r) = \delta_2[\underbrace{1,\ldots,1}_{2^{r-i}}]$ for $p \in \mathbf{p} := \{1,3,\ldots,2^i-1\}$ and $\mathrm{Blk}_q(S_i^r) = \delta_2[\underbrace{2,\ldots,2}_{2^{r-i}}]$ for $q \in \mathbf{q} := \{2,4,\ldots,2^i\}$.

The algebraic representation (12b) shows that $x^2(t+1)$ is not affected by the control $u(t)$, hence one just need to consider $x^1(t+1)$. Note that $S_i^r x^1 = x_i$ $(i = 1,\ldots,r)$, from (12a) we obtain

$$x_i(t+1) = S_i^r G_1 u(t)x(t)\cdots x(t-\tau) \triangleq \tilde{G}_i u(t)x(t)\cdots x(t-\tau), \tag{16}$$

where $\tilde{G}_i = S_i^r G_1 \in \mathcal{L}_{2\times 2^{n(\tau+1)+r}}$. If the component-wise algebraic form (16) of the network (11) has the same form with (4a), then there exists matrix $\hat{G}_i \in \mathcal{L}_{2\times 2^{n(\tau+1)+1}}$ such that

$$x_i(t+1) = \tilde{G}_i u(t)x(t)\cdots x(t-\tau) = \hat{G}_i u_i(t)x(t)\cdots x(t-\tau) = \hat{G}_i S_i^r u(t)x(t)\cdots x(t-\tau). \tag{17}$$

Since $\hat{G}_i$ $(i = 1,\ldots,r)$ are $2\times 2^{n(\tau+1)+1}$ logical matrices, we split each of them into 2 equal blocks as $\hat{G}_i = [\hat{G}_{i1}\ \hat{G}_{i2}]$, where $\hat{G}_{ik} = \mathrm{Blk}_k(\hat{G}_i) \in \mathcal{L}_{2\times 2^{n(\tau+1)}}$, $k = 1,2$. It follows from the above equation that

$$\begin{aligned} \tilde{G}_i = \hat{G}_i S_i^r &= [\hat{G}_{i1}\ \hat{G}_{i2}]\delta_2[\underbrace{1,\ldots,1}_{2^{r-i}},\underbrace{2,\ldots,2}_{2^{r-i}},\ldots,\underbrace{1,\ldots,1}_{2^{r-i}},\underbrace{2,\ldots,2}_{2^{r-i}}] \\ &= [\underbrace{\hat{G}_{i1}\ \cdots\ \hat{G}_{i1}}_{2^{r-i}}\ \underbrace{\hat{G}_{i2}\ \cdots\ \hat{G}_{i2}}_{2^{r-i}}\ \cdots\ \underbrace{\hat{G}_{i1}\ \cdots\ \hat{G}_{i1}}_{2^{r-i}}\ \underbrace{\hat{G}_{i2}\ \cdots\ \hat{G}_{i2}}_{2^{r-i}}] \\ &\triangleq [\mathrm{Blk}_1(\tilde{G}_i)\ \mathrm{Blk}_2(\tilde{G}_i)\ \cdots\ \mathrm{Blk}_{2^i}(\tilde{G}_i)], \end{aligned} \tag{18}$$

which implies the following result about the inherent special structure of the structure matrix $\tilde{G}_i$.

Theorem 2. Let (13) be the algebraic representation of ABCN (11), where G is in the form of (8) (i.e., $r = s$). Then one can obtain matrices $G_1 \in \mathcal{L}_{2^s\times 2^{n(\tau+1)+r}}$, $G_2 \in \mathcal{L}_{2^{n-s}\times 2^{n(\tau+1)}}$, and $\tilde{G}_i \in \mathcal{L}_{2\times 2^{n(\tau+1)+r}}$, $i = 1,\ldots,r$, satisfying $G = G_1 * (\mathbf{1}_{2^r}^{\mathrm{T}} \otimes G_2)$ and $G_1 = \tilde{G}_1 * \cdots * \tilde{G}_r$. Furthermore, if there exists matrix $\hat{G}_i = [\hat{G}_{i1}\ \hat{G}_{i2}] \triangleq [\mathrm{Blk}_1(\hat{G}_i)\ \mathrm{Blk}_2(\hat{G}_i)]$ such that $\mathrm{Blk}_p(\tilde{G}_i) = \mathbf{1}_{2^{r-i}}^{\mathrm{T}} \otimes \hat{G}_{i1}$ for $p \in \mathbf{p}$ and $\mathrm{Blk}_q(\tilde{G}_i) = \mathbf{1}_{2^{r-i}}^{\mathrm{T}} \otimes \hat{G}_{i2}$ for $q \in \mathbf{q}$, which $i = 1,\ldots,r$, then the dynamic of the time delay BCN (11) is in the form of (3).

Remark 2. In fact, if the matrix $\tilde{G}_i$ is in the form of (18), then one can easily derive the component-wise algebraic form of ABCN (3) as follows:

$$\begin{cases} x_1(t+1) = \hat{G}_1 u_1(t)x(t)\cdots x(t-\tau), \\ \qquad\vdots \\ x_r(t+1) = \hat{G}_r u_r(t)x(t)\cdots x(t-\tau), \\ x_{r+1}(t+1) = \hat{G}_{r+1} x(t)\cdots x(t-\tau), \\ \qquad\vdots \\ x_n(t+1) = \hat{G}_n x(t)\cdots x(t-\tau), \end{cases} \tag{19}$$

where $\hat{G}_{r+1} * \cdots * \hat{G}_n = G_2$. Based on (19), one can reconstruct the ABCN (3) in its logical form by resorting to the method proposed in [8].

Example 2. Suppose that the algebraic representation of an ABCN is

$$x(t+1) = Gu(t)x(t)x(t-1)x(t-2),$$

where $x = \ltimes_{k=1}^{3} x_k$, $u = u_1u_2$, and the network transition matrix G is given the same as presented in (10). Then one has matrices $G_1 = F_1$ and $G_2 = F_2$ satisfying $G = G_1 * (\mathbf{1}_{2^2}^{\mathrm{T}} \otimes G_2)$. Furthermore, by calculation, we can find two matrices $\tilde{G}_1$ and $\tilde{G}_2$ such that $G_1 = \tilde{G}_1 * \tilde{G}_2$:

$$\begin{cases} \tilde{G}_1 = \delta_2[\underbrace{1,1,\ldots,2}_{2^9}, \vdots \underbrace{1,1,\ldots,2}_{2^9}, \vdots \underbrace{2,2,\ldots,1}_{2^9}, \vdots \underbrace{2,2,\ldots,1}_{2^9}] \triangleq [\hat{G}_{11}\ \hat{G}_{11}\ \hat{G}_{12}\ \hat{G}_{12}], \\ \tilde{G}_2 = \delta_2[\underbrace{1,1,\ldots,1}_{2^9}, \vdots \underbrace{1,2,\ldots,2}_{2^9}, \vdots \underbrace{1,1,\ldots,1}_{2^9}, \vdots \underbrace{1,2,\ldots,2}_{2^9}] \triangleq [\hat{G}_{21}\ \hat{G}_{22}\ \hat{G}_{21}\ \hat{G}_{22}]. \end{cases}$$

Therefore, one can conclude that the dynamic of this ABCN is in the form of (3) by Theorem 2. Moreover, the dynamical model of this ABCN can be constructed in its logical form as (9).

3.3 Equivalent form of the ABCNs

Denote $z_k(t) = x(t-k)$ $(k = 0, 1, \ldots, \tau)$ and $z = \ltimes_{k=0}^{\tau} z_k$, the algebraic form (7) can be converted into the form

$$x(t+1) = Fu(t)z(t).$$

This, together with Lemma 3, yields

$$\begin{aligned} z(t+1) &= x(t+1)x(t)\cdots x(t-\tau+1) \\ &= (I_{2^{n(\tau+1)}} \otimes \mathbf{1}_{2^n}^{\mathrm{T}})x(t+1)x(t)\cdots x(t-\tau+1)x(t-\tau) \\ &= (I_{2^{n(\tau+1)}} \otimes \mathbf{1}_{2^n}^{\mathrm{T}})Fu(t)\Psi_{2^{n(\tau+1)}}z(t) \\ &= (I_{2^{n(\tau+1)}} \otimes \mathbf{1}_{2^n}^{\mathrm{T}})F(I_{2^r} \otimes \Psi_{2^{n(\tau+1)}})u(t)z(t) \\ &\triangleq Lu(t)z(t), \end{aligned} \tag{20}$$

where $L = (I_{2^{n(\tau+1)}} \otimes \mathbf{1}_{2^n}^{\mathrm{T}})F(I_{2^r} \otimes \Psi_{2^{n(\tau+1)}}) \in \mathcal{L}_{2^{n(\tau+1)} \times 2^{n(\tau+1)+r}}$.

In fact, the algebraic representation (7) on x can also be obtained from the algebraic representation (20) on z. In the following, we will explain this fact.

Note that $x(t+1) = (I_{2^n} \otimes \mathbf{1}_{2^{n\tau}}^{\mathrm{T}})z(t+1)$, then by the algebraic representation (20), one obtains the algebraic representation on x as follows:

$$x(t+1) = (I_{2^n} \otimes \mathbf{1}_{2^{n\tau}}^{\mathrm{T}})Lu(t)x(t)\cdots x(t-\tau). \tag{21}$$

Our task now is to verify that

$$(I_{2^n} \otimes \mathbf{1}_{2^{n\tau}}^{\mathrm{T}})L = F. \tag{22}$$

Applying the properties of the Kronecker product yields

$$\begin{aligned} (I_{2^n} \otimes \mathbf{1}_{2^{n\tau}}^{\mathrm{T}})L &= (I_{2^n} \otimes \mathbf{1}_{2^{n\tau}}^{\mathrm{T}})(I_{2^{n(\tau+1)}} \otimes \mathbf{1}_{2^n}^{\mathrm{T}})F(I_{2^r} \otimes \Psi_{2^{n(\tau+1)}}) \\ &= \left\{I_{2^n} \otimes [\mathbf{1}_{2^{n\tau}}^{\mathrm{T}}(I_{2^{n\tau}} \otimes \mathbf{1}_{2^n}^{\mathrm{T}})]\right\}F(I_{2^r} \otimes \Psi_{2^{n(\tau+1)}}) \\ &= (I_{2^n} \otimes \mathbf{1}_{2^{n(\tau+1)}}^{\mathrm{T}})(F \otimes I_{2^{n(\tau+1)}})(I_{2^r} \otimes \Psi_{2^{n(\tau+1)}}) \\ &= (F \otimes \mathbf{1}_{2^{n(\tau+1)}}^{\mathrm{T}})(I_{2^r} \otimes \Psi_{2^{n(\tau+1)}}). \end{aligned} \tag{23}$$

Since F in (7) is a $2^n \times 2^{n(\tau+1)+r}$ logical matrix, we split it into 2^r square blocks as $F = [F_1\ F_2\ \cdots\ F_{2^r}]$, where $F_k \in \mathcal{L}_{2^n \times 2^{n(\tau+1)}}$, $k = 1, \ldots, 2^r$. Then, we have

$$\begin{aligned}(F \otimes \mathbf{1}^{\mathrm{T}}_{2^{n(\tau+1)}})(I_{2^r} \otimes \Psi_{2^{n(\tau+1)}}) &= [F_1 \otimes \mathbf{1}^{\mathrm{T}}_{2^{n(\tau+1)}}\ \cdots\ F_{2^r} \otimes \mathbf{1}^{\mathrm{T}}_{2^{n(\tau+1)}}] \begin{bmatrix} \Psi_{2^{n(\tau+1)}} & & \\ & \ddots & \\ & & \Psi_{2^{n(\tau+1)}} \end{bmatrix} \\ &= [(F_1 \otimes \mathbf{1}^{\mathrm{T}}_{2^{n(\tau+1)}})\Psi_{2^{n(\tau+1)}}\ \cdots\ (F_{2^r} \otimes \mathbf{1}^{\mathrm{T}}_{2^{n(\tau+1)}})\Psi_{2^{n(\tau+1)}}] \\ &= [F_1\ \cdots\ F_{2^r}].\end{aligned}$$

This together with (23) implies that (22) holds.

According to the above analysis, one can conclude that there is a one-to-one and onto mapping between (7) and (20). Therefore, the reachability as well as controllability of ABCN (3) can be investigated directly from system (20) instead of system (7).

3.4 Controllability of the ABCNs

In what follows, the controllability of ABCNs (3) is addressed based on the algebraic form (20). We first present the definition of controllability for network (3).

Definition 3. Consider system (3). For any given initial time t_0, any given set of time delays $\{\tau_{ij} \in \mathcal{N} : i, j = 1, \ldots, n\}$, any given initial state sequence $X_0 = \{x(t_0 - \tau), x(t_0 - \tau + 1), \ldots, x(t_0)\} \sim x_0 = \ltimes_{k=0}^{\tau} x(t_0 - k) \in \Delta_{2^{n(\tau+1)}}$, any given destination state $x_d \in \Delta_{2^n}$, and any given $k \in \mathcal{N} \setminus \{0\}$.

(1) x_d is said to be reachable from x_0 at the kth step if a control sequence $\{u(t_0), u(t_0+1), \ldots, u(t_0 + k - 1)\}$ can be found such that the trajectory of (3) satisfies $x(t_0 + k) = x_d$.

(2) The set of all states that are reachable from x_0 at the kth step is said to be the k-step reachable set of x_0, denoted by $R_k(x_0)$.

(3) The set of all states that are reachable from x_0 is called to be the reachable set of x_0, denoted by $R(x_0)$. Clearly $R(x_0) = \cup_{k \in \mathcal{N} \setminus \{0\}} R_k(x_0)$.

(4) System (3) is said to be controllable from x_0 if $R(x_0) = \Delta_{2^n}$.

(5) System (3) is said to be (globally) controllable if it is controllable from any $x_0 \in \Delta_{2^{n(\tau+1)}}$.

For $x_0 \in \Delta_{2^{n(\tau+1)}}$ and $x_d \in \Delta_{2^n}$, let $l(k; x_0, x_d)$ denote the number of different control sequences that steer ABCN (3) from x_0 to x_d at the kth time-step. The following result provides a simple algebraic expression for $l(k; x_0, x_d)$.

Theorem 3. Consider system (3) with algebraic form (20). Let $l(k; x_0, x_d)$ denote the number of different control sequences that steer ABCN (3) from $x_0 = \delta^j_{2^{n(\tau+1)}}$ to $x_d = \delta^i_{2^n}$ at the kth time-step. Then

$$l(k; x_0, x_d) = \left[(I_{2^n} \otimes \mathbf{1}^{\mathrm{T}}_{2^{n\tau}})M^k\right]_{ij}, \tag{24}$$

where $M = L\mathbf{1}_{2^r}$.

Proof. We prove this result by induction. Consider firstly the case $k = 1$. For a given initial state x_0, it follows from (21) that

$$x_d = x(t_0 + 1) = (I_{2^n} \otimes \mathbf{1}^{\mathrm{T}}_{2^{n\tau}})L \ltimes u(t_0) \ltimes x_0. \tag{25}$$

Let $\mu^1(t_0), \ldots, \mu^\alpha(t_0)$ be the different control sequences steering (25) from x_0 to x_d, i.e.,

$$x_d = (I_{2^n} \otimes \mathbf{1}^{\mathrm{T}}_{2^{n\tau}})L \ltimes \mu^i(t_0) \ltimes x_0, \quad i \in \{1, \ldots, \alpha\}. \tag{26}$$

Since each value of the controllers is a column of I_{2^r}, there exist $\beta = 2^r - \alpha$ different control sequences $\{\nu^j(t_0)\}$ such that

$$x_d \neq (I_{2^n} \otimes \mathbf{1}^{\mathrm{T}}_{2^{n\tau}})L \ltimes \nu^j(t_0) \ltimes x_0, \quad j \in \{1, \ldots, \beta\}. \tag{27}$$

Multiplying both sides of (26) and (27) with x_d^{T} yields

$$1 = x_d^{\mathrm{T}}(I_{2^n} \otimes \mathbf{1}^{\mathrm{T}}_{2^{n\tau}})L \ltimes \mu^i(t_0) \ltimes x_0, \quad i \in \{1, \ldots, \alpha\},$$

$$0 = x_d^{\mathrm{T}}(I_{2^n} \otimes \mathbf{1}_{2^{n\tau}}^{\mathrm{T}})L \ltimes \nu^j(t_0) \ltimes x_0, \quad j \in \{1, \ldots, \beta\}.$$

Summing up the above 2^r equations together, we have

$$\alpha = x_d^{\mathrm{T}}(I_{2^n} \otimes \mathbf{1}_{2^{n\tau}}^{\mathrm{T}})L \ltimes \mathbf{1}_{2^r} \ltimes x_0 = x_d^{\mathrm{T}}(I_{2^n} \otimes \mathbf{1}_{2^{n\tau}}^{\mathrm{T}})Mx_0 = [(I_{2^n} \otimes \mathbf{1}_{2^{n\tau}}^{\mathrm{T}})M]_{ij}, \tag{28}$$

which means that (24) holds for $k = 1$.

Secondly, assume that (24) holds for $k = n$. For the induction step, we consider finally the case of $k = n + 1$. Note that the number of control sequences that steer system (3) from $x_0 = z(t_0)$ to x_d at $n + 1$ time-step equals to the sum, over all possible states $z(t_0 + 1) = \delta_{2^{n(\tau+1)}}^{\lambda}$, of the product of (i) the number of control sequences that steer system (3) from $z(t_0)$ to $z(t_0 + 1)$ at one step; and (ii) the number of control sequences that steer system (3) from $z(t_0 + 1)$ to x_d at n steps. It thus follows that

$$\begin{aligned} l(n+1; x_0, x_d) &= \sum_{\lambda=1}^{2^{n(\tau+1)}} l(1; z(t_0), z(t_0+1)) l(n; z(t_0+1), x_d) \\ &= \sum_{\lambda=1}^{2^{n(\tau+1)}} l(n; z(t_0+1), x_d) l(1; z(t_0), z(t_0+1)). \end{aligned} \tag{29}$$

Applying the induction hypothesis yields

$$\begin{aligned} l(n+1; x_0, x_d) &= \sum_{\lambda=1}^{2^{n(\tau+1)}} x_d^{\mathrm{T}}[(I_{2^n} \otimes \mathbf{1}_{2^{n\tau}}^{\mathrm{T}})M^n] z(t_0+1) z(t_0+1)^{\mathrm{T}} M z(t_0) \\ &= \sum_{\lambda=1}^{2^{n(\tau+1)}} [(I_{2^n} \otimes \mathbf{1}_{2^{n\tau}}^{\mathrm{T}})M^n]_{i\lambda} M_{\lambda j} \\ &= [(I_{2^n} \otimes \mathbf{1}_{2^{n\tau}}^{\mathrm{T}})M^{n+1}]_{ij}, \end{aligned} \tag{30}$$

which infers that (24) holds for $k = n+1$. By induction, one can conclude that (24) holds for any positive integer k. The proof is then complete.

In fact, the k in Definition 3 about reachability and controllability depends on both x_0 and x_d. Note that the cardinal number of the state space $\Delta_{2^{n(\tau+1)}}$ is $2^{n(\tau+1)}$, one can choose $k(x_0, x_d) \leqslant 2^{n(\tau+1)} - 1 := N$. Based on Theorem 3, the following result on testing the controllability of (3) is obtained.

Theorem 4. Consider system (3) with algebraic form (20). Let $N = 2^{n(\tau+1)} - 1$, then

(1) $x_d = \delta_{2^n}^i$ is reachable from $x_0 = \delta_{2^{n(\tau+1)}}^j$ at the kth step if and only if

$$\left[(I_{2^n} \otimes \mathbf{1}_{2^{n\tau}}^{\mathrm{T}})M^k\right]_{ij} > 0.$$

(2) System (3) is controllable from $x_0 = \delta_{2^{n(\tau+1)}}^j$ if and only if

$$\mathrm{Col}_j\big((I_{2^n} \otimes \mathbf{1}_{2^{n\tau}}^{\mathrm{T}}) \sum_{\lambda=1}^{N} M^{\lambda}\big) > 0.$$

(3) System (3) is controllable if and only if

$$(I_{2^n} \otimes \mathbf{1}_{2^{n\tau}}^{\mathrm{T}}) \sum_{\lambda=1}^{N} M^{\lambda} > 0.$$

Remark 3. Theorem 4 provides some necessary and sufficiency conditions for the reachability and controllability of ABCN (3) with pinning controllers, and the obtained matrix testing criteria imply that one only needs to check for a finite number of N. One can observe that the matrix M defined in Theorem 3 is a $2^{n(\tau+1)} \times 2^{n(\tau+1)}$ matrix, which means that the dimensions of M grows exponentially as the size of the network (3) increases. Thus, the proposed criteria are applicable only to the small scale of ABCNs.

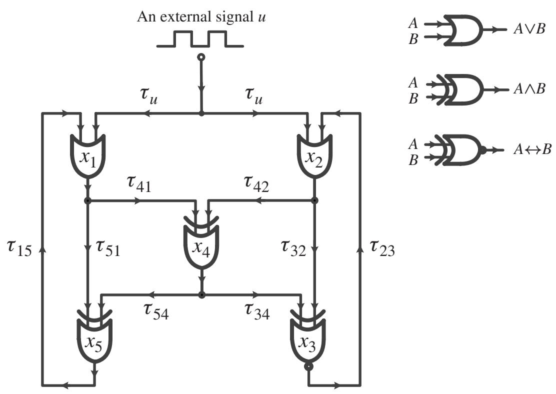

Figure 1 Schematic of the network topology for the five-node ABCN on the IEEE Std 91/91a-1991 representation [33].

Table 1 Time delays τ_{ij} in system (31)

τ_u	τ_{15}	τ_{23}	τ_{32}	τ_{34}	τ_{41}	τ_{42}	τ_{51}	τ_{54}
1	2	1	1	1	2	2	1	2

Example 3. In [33], the authors have proposed the topology of an ABN that is driven by an external signal u as shown in Figure 1. The system is an electronic circuit that realizes the Boolean nodes with logic gates, specifically, nodes 1 and 2 execute the OR logic operation, node 3 executes the XNOR logic operation, while nodes 4 and 5 execute the XOR logic operation. The time that it takes a signal to propagate to node i from node j is denoted by τ_{ij} $(1 \leqslant i, j \leqslant 5)$. Each time delay comes about from a combination of an intrinsic delay associated with each gate and the signal propagation time along the connecting links. The Boolean delay equations describing this ABCN are as follows:

$$\begin{cases} x_1(t) = u(t-\tau_u) \vee x_5(t-\tau_{15}), \\ x_2(t) = u(t-\tau_u) \vee x_3(t-\tau_{23}), \\ x_3(t) = x_2(t-\tau_{32}) \leftrightarrow x_4(t-\tau_{34}), \\ x_4(t) = x_1(t-\tau_{41}) \wedge x_2(t-\tau_{42}), \\ x_5(t) = x_1(t-\tau_{51}) \wedge x_4(t-\tau_{54}), \end{cases} \tag{31}$$

where the values of τ_{ij} are given in Table 1.

Let $x = \ltimes_{k=1}^{5} x_k$, then we can express the ABCN (31) in its algebraic form as

$$x(t+1) = Fu(t)x(t)x(t-1), \tag{32}$$

where F is a $2^5 \times 2^{11}$ logical matrix which is omitted here for space consideration, and $t = 0, 1, 2, \ldots$. Set $z_0(t) = x(t)$ and $z_1(t) = x(t-1)$, the equivalent algebraic form of ABCN (32) can be obtained as $z(t+1) = Lu(t)z(t)$, where $L \in \mathcal{L}_{2^{10} \times 2^{11}}$. In order to check whether or not the ABCN (31) is controllable, we should calculate

$$(I_{2^n} \otimes \mathbf{1}_{2^{n\tau}}^{\mathrm{T}}) \sum_{\lambda=1}^{2^{10}-1} M^{\lambda} := [\Gamma_1\ \Gamma_2\ \Gamma_3\ \Gamma_4] \in \mathbb{R}^{32 \times 1024}$$

according to Theorem 4. By choosing $\bar{N} = 20$, and replacing the nonzero entries of $\Gamma_k \in \mathbb{R}^{32 \times 256}$ $(k = 1, \ldots, 4)$ by 1, we obtain

$$\Gamma_1 = \Gamma_2 = \Gamma_3 = \Gamma_4 > 0,$$

and Figure 2 plots the whole row indexes of each column of matrix Γ_k. Thus, one can conclude that the ABCN (31) is globally controllable.

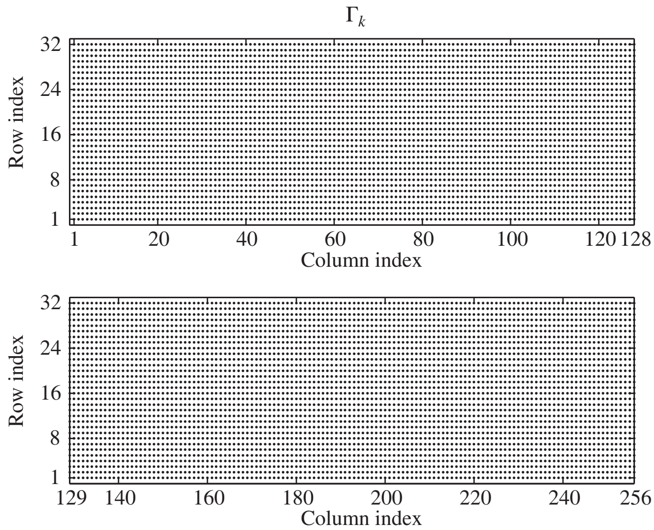

Figure 2 The whole row indexes of each column of matrix Γ_k. Each point corresponds to the row index of each column, which implies the position of 1.

3.5 Pinning control design algorithm

Consider the problem of designing a pinning control sequence that steers the ABCN (3) between any two given states x_0 and x_d, which seems relevant to the therapeutic intervention, since some states may correspond to the diseased states.

First, for system (20), if the two states $z_d = \delta^{i'}_{2^{n(\tau+1)}}$ and $z(t_0) = \delta^{j}_{2^{n(\tau+1)}}$ satisfy $\delta^{i'}_{2^{n(\tau+1)}} \in R(\delta^{j}_{2^{n(\tau+1)}})$, Algorithm 1 can generate any control sequence steering $z(t_0)$ to z_d.

Algorithm 1 Summary of computational procedure for designing a control sequence that steers $z(t_0)$ to z_d

Require: $[M^\kappa]_{i'j} > 0$ holds for some κ;

1: Find one (or the smallest) κ such that $\left[M^\kappa\right]_{i'j} > 0$, set $z(t_0) = \delta^{j}_{2^{n(\tau+1)}}$ and $z(t_0+\kappa) = \delta^{i'}_{2^{n(\tau+1)}}$, go to Step 2;

2: If $\kappa = 1$, find one μ such that $[L\delta^{\mu}_{2^r}]_{i'j} > 0$, set $u(t_0+\kappa-1) = \delta^{\mu}_{2^r}$, stop. Else, find one λ such that $[M]_{i'\lambda} > 0$ and $[M^{\kappa-1}]_{\lambda j} > 0$, set $z(t_0+\kappa-1) = \delta^{\lambda}_{2^{n(\tau+1)}}$; find one μ such that $[L\delta^{\mu}_{2^r}]_{i'\lambda} > 0$, set $u(t_0+\kappa-1) = \delta^{\mu}_{2^r}$;

3: Set $\kappa = \kappa - 1$, $i' = \lambda$, go back to Step 2.

Next, consider ABCN (3) with its algebraic form (20). Suppose that the two given states $x_0 = \delta^{j}_{2^{n(\tau+1)}}$ and $x_d = \delta^{i}_{2^n}$ satisfy $x_d \in R(x_0)$, then by Theorem 4, one has

$$\left[(I_{2^n} \otimes \mathbf{1}^{\mathrm{T}}_{2^{n\tau}})M^\kappa\right]_{ij} > 0$$

holds for some κ, which implies that

$$\mathrm{Row}_i(I_{2^n} \otimes \mathbf{1}^{\mathrm{T}}_{2^{n\tau}})\mathrm{Col}_j(M^\kappa) > 0. \tag{33}$$

Thus, there exists at least one $i' \in \{(i-1)2^{n\tau} + l : l = 1, 2, \ldots, 2^{n\tau}\}$ such that $[M^\kappa]_{i'j} > 0$, and we have

$$\delta^{i}_{2^n} = (I_{2^n} \otimes \mathbf{1}^{\mathrm{T}}_{2^{n\tau}})\delta^{i'}_{2^{n(\tau+1)}}.$$

Then, by Algorithm 1, one can find a control sequence $\{u(t_0), u(t_0+1), \ldots, u(t_0+k-1)\}$ steering $\delta^{j}_{2^{n(\tau+1)}}$ to $\delta^{i'}_{2^{n(\tau+1)}}$, thereby providing the pinning control sequence steering the ABCN (3) from $x_0 = \delta^{j}_{2^{n(\tau+1)}}$ to $x_d = \delta^{i}_{2^n}$.

Finally, based on the above analysis, we give the following pinning control design algorithm for system (3). It is worth pointing out that the pinning controller form x_0 to x_d is generally not unique, Algorithm 2 gives a unified method to find all pinning controllers, including the shortest ones.

Remark 4. The time delays in ABCN (3) are all assumed to be constant, and it is possible to extend the present study to the case of time-varying delays by splitting the system into a finite number of subsystems with no time delays, called the constructed forest. The readers are referred to [26] for more details.

Algorithm 2 Summary of computational procedure for designing a control sequence that steers x_0 to x_d

Require: $\left[(I_{2^n} \otimes \mathbf{1}^{\mathrm{T}}_{2^{n\tau}})M^{\kappa}\right]_{ij} > 0$ holds for some κ;

1: Find one (or the smallest) κ such that $\left[M^{\kappa}\right]_{i'j} > 0$ holds for some $i' \in \{(i-1)2^{n\tau} + l : l = 1, 2, \ldots, 2^{n\tau}\}$. Set $z(t_0) = \delta^{j}_{2^{n(\tau+1)}}$, $z(t_0+\kappa) = \delta^{i'}_{2^{n(\tau+1)}}$, go to Step 2;

2: If $\kappa = 1$, find one μ such that $[L\delta^{\mu}_{2^r}]_{i'j} > 0$, set $u(t_0+\kappa-1) = \delta^{\mu}_{2^r}$, stop. Else, find one λ such that $[M]_{i'\lambda} > 0$, $[M^{\kappa-1}]_{\lambda j} > 0$, set $z(t_0+\kappa-1) = \delta^{\lambda}_{2^{n(\tau+1)}}$; find one μ such that $[L\delta^{\mu}_{2^r}]_{i'\lambda} > 0$, set $u(t_0+\kappa-1) = \delta^{\mu}_{2^r}$;

3: Set $\kappa = \kappa - 1$, $i' = \lambda$, go back to Step 2.

Example 4. Consider the ABCN in Example 3, and design a control sequence to steer the initial state $x_0 = \delta^{242}_{1024}$ to the destination state $x_d = \delta^{19}_{32}$ (if possible). We follow Algorithm 2 step by step as follows:

- Step 1. A calculation yields $\left[M^3\right]_{603,242} > 0$ with $l = 27$. Set $z(0) = \delta^{242}_{1024}$, $z(3) = \delta^{603}_{1024}$.
- Step 2. From a straightforward computation, one gets $[M]_{603,838} > 0$, $[M^2]_{838,242} > 0$, set $z(2) = \delta^{838}_{1024}$. One also gets $[L\delta^2_2]_{603,838} > 0$, set $u(2) = \delta^2_2$.
- Step 3. From a straightforward computation, one gets $[M]_{838,168} > 0$, $[M^1]_{168,242} > 0$, set $z(1) = \delta^{168}_{1024}$. One also gets $[L\delta^2_2]_{838,168} > 0$, set $u(1) = \delta^2_2$.
- Step 4. From a straightforward computation, one gets $[L\delta^1_2]_{168,242} > 0$, set $u(0) = \delta^1_2$.

Consequently, one obtains the control sequence $\{u(0) = \delta^1_2, u(1) = \delta^2_2, u(2) = \delta^2_2\}$ steering ABCN (31) from $x_0 = \delta^{242}_{1024}$ to $x_d = \delta^{19}_{32}$ at the third step. Since $z(1) = \delta^{168}_{1024}$, $z(2) = \delta^{838}_{1024}$, the corresponding state trajectory is

$$\left.\begin{cases} x(0) = \delta^{8}_{32} \\ x(-1) = \delta^{18}_{32} \end{cases}\right\} \xrightarrow{u(0)=\delta^1_2} x(1) = \delta^{6}_{32} \xrightarrow{u(1)=\delta^2_2} x(2) = \delta^{27}_{32} \xrightarrow{u(2)=\delta^2_2} x(3) = \delta^{19}_{32}. \tag{34}$$

4 Conclusion

In this paper, the pinning controllability of the ABCNs has been studied, and several new results have been presented. By resorting to the STP technique, ABCNs with pinning controllers have been converted into the algebraic form, and some inherent special structures of the network transition matrix have been characterized. Then, a simple algebraic formula has been obtained for the number of different control sequences that steer an ABCN from the given initial state to the desired terminal state at a given number of time step. With this expression, some computable algebraic criteria have been derived for the pinning controllability of the ABCNs. Moreover, a practical control method has been devised that can be implemented to only a fraction of nodes to force the whole system to the desired state. A practical ABCN realized by logic circuits has been given to highlight the utility of the obtained results.

On the other hand, the limitation of our method lies in the fact that the state space grows exponentially as the size of the network increases. Further research is needed to apply the obtained results to the large-scale ABCNs. Another interesting yet difficult task in future work is to determine the potential nodes selected to be controlled. In summary, the proposed approach provides new insights on understanding and controlling the dynamics of GRNs by means of ABCNs, and it also has implications for the therapeutic intervention and in the genetic engineering.

Acknowledgements This work was supported in part by National Natural Science Foundation of China (Grant Nos. 61329301, 61273156), Natural Science Foundation of Jiangsu Province of China (Grant No. BK20130017), Six Talent Peaks Project for the High Level Personnel from the Jiangsu Province of China (Grant No. 2015-DZXX-003), Scientific Research Foundations of Graduate School of Southeast University (Grant No. YBJJ1560), and Innovation Program of Jiangsu Province (Grant No. KYZZ15_0050).

Conflict of interest The authors declare that they have no conflict of interest.

References

1 Wang Z D, Wu H H, Liang J L, et al. On modeling and state estimation for genetic regulatory networks with polytopic uncertainties. IEEE Trans Nanobiosci, 2013, 12: 13–20

2 Shen B, Wang Z D, Liang J L, et al. Sampled-data H_∞ filtering for stochastic genetic regulatory networks. Int J Robust Nonlinear Control, 2011, 21: 1759–1777

3 Kauffman S A. Metabolic stability and epigenesis in randomly constructed genetic nets. J Theor Biol, 1969, 22: 437–467

4 Aldana M. Boolean dynamics of networks with scale-free topology. Phys D, 2003, 185: 45–66

5 Akutsu T, Hayashida M, Ching W-K, et al. Control of Boolean networks: hardness results and algorithms for tree structured networks. J Theor Biol, 2007, 244: 670–679

6 Ideker T, Galitski T, Hood L. A new approach to decoding life: systems biology. Annu Rev Genomics Hum Genet, 2001, 2: 343–372

7 Cheng D Z, Qi H S. A linear representation of dynamics of Boolean networks. IEEE Trans Automat Control, 2010, 55: 2251–2258

8 Cheng D Z, Qi H S. Controllability and observability of Boolean control networks. Automatica, 2009, 45: 1659–1667

9 Laschov D, Margaliot M. Controllability of Boolean control networks via the Perron-Frobenius theory. Automatica, 2012, 48: 1218–1223

10 Fornasini E, Valcher M E. Observability, reconstructibility and state observers of Boolean control networks. IEEE Trans Automat Control, 2013, 58: 1390–1401

11 Cheng D Z, Qi H S, Li Z Q, et al. Stability and stabilization of Boolean networks. Int J Robust Nonlinear Control, 2011, 21: 134–156

12 Li H T, Wang Y Z. Consistent stabilizability of switched Boolean networks. Neural Netw, 2013, 46: 183–189

13 Li H T, Wang Y Z, Liu Z B. Stability analysis for switched Boolean networks under arbitrary switching signals. IEEE Trans Automat Control, 2014, 59: 1978–1982

14 Zhao Y, Cheng D Z. On controllability and stabilizability of probabilistic Boolean control networks. Sci China Inf Sci, 2014, 57: 012202

15 Fornasini E, Valcher M E. Optimal control of Boolean control networks. IEEE Trans Automat Control, 2014, 59: 1258–1270

16 Laschov D, Margaliot M. A maximum principle for single-input Boolean control networks. IEEE Trans Automat Control, 2011, 56: 913–917

17 Zou Y L, Zhu J D. System decomposition with respect to inputs for Boolean control networks. Automatica, 2014, 50: 1304–1309

18 Zou Y L, Zhu J D. Kalman decomposition for Boolean control networks. Automatica, 2015, 54: 65–71

19 Cheng D Z. On finite potential games. Automatica, 2014, 50: 1793–1801

20 Xu X R, Hong Y G. Matrix approach to model matching of asynchronous sequential machines. IEEE Trans Automat Control, 2013, 58: 2974–2979

21 Smolen P, Baxter D A, Byrne J H. Mathematical modeling of gene networks. Neuron, 2000, 26: 567–580

22 Li F F, Sun J T. Controllability of Boolean control networks with time delays in states. Automatica, 2011, 47: 603–607

23 Li F F, Sun J T. Controllability of higher order Boolean control networks. Appl Math Comput, 2012, 219: 158–169

24 Chen H, Sun J T. A new approach for global controllability of higher order Boolean control network. Neural Netw, 2013, 39: 12–17

25 Zhang L J, Zhang K Z. Controllability of time-variant Boolean control networks and its application to Boolean control networks with finite memories. Sci China Inf Sci, 2013, 56: 108201

26 Zhang L J, Zhang K Z. Controllability and observability of Boolean control networks with time-variant delays in states. IEEE Trans Neural Netw Learn Syst, 2013, 24: 1478–1484

27 Müller F-J, Schuppert A. Few inputs can reprogram biological networks. Nature, 2011, 478: E4

28 Lin G-Q, Ao B, Chen J-W, et al. Modeling and controlling the two-phase dynamics of the p53 network: a Boolean network approach. New J Phys, 2014, 16: 125010

29 Lu J Q, Zhong J, Huang C, et al. On pinning controllability of Boolean control networks. IEEE Trans Automat Control, 2015, 61: 1658–1663

30 Li F F. Pinning control design for the stabilization of Boolean networks. IEEE Trans Neural Netw Learn Syst, 2015, doi: 10.1109/TNNLS.2015.2449274

31 Ghil M, Zaliapin I, Coluzzi B. Boolean delay equations: a simple way of looking at complex systems. Phys D, 2008, 237: 2967–2986

32 Zhang R, Cavalcante H L, Gao Z, et al. Boolean chaos. Phys Rev E, 2009, 80: 045202

33 Rivera-Durón R R, Campos-Cantón E, Campos-Cantón I, et al. Forced synchronization of autonomous dynamical Boolean networks. Chaos, 2015, 25: 083113

34 Glass L, Perkins T J, Mason J, et al. Chaotic dynamics in an electronic model of a genetic network. J Stat Phys, 2005, 121: 969–994

35 Cheng X R, Sun M Y, Socolar J E S. Autonomous Boolean modelling of developmental gene regulatory networks. J R Soc Interface, 2013, 10: 20120574