Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Output tracking of delayed logical control networks with multi-constraint*

Ya-ting ZHENG, Jun-e FENG†‡

School of Mathematics, Shandong University, Jinan 250100, China

†E-mail: fengjune@sdu.edu.cn

Received July 29, 2019; Revision accepted Sept. 21, 2019; Crosschecked Dec. 19, 2019

Abstract: In this study, the output tracking of delayed logical control networks (DLCNs) with state and control constraints is further investigated. Compared with other delays, state-dependent delay updates its value depending on the current state values and a pseudo-logical function. Multiple constraints mean that state values are constrained in a nonempty set and the design of the controller is conditioned. Using the semi-tensor product of matrices, dynamical equations of DLCNs are converted into an algebraic description, and an equivalent augmented system is constructed. Based on the augmented system, the output tracking problem is transformed into a set stabilization problem. A deformation of the state transition matrix is computed, and a necessary and sufficient condition is derived for the output tracking of a DLCN with multi-constraint. This condition is easily verified by mathematical software. In addition, the admissible state-feedback controller is designed to enable the outputs of the DLCN to track the reference signal. Finally, theoretical results are illustrated by an example.

Key words: Logical control networks; Multi-constraint; Output tracking; Stabilization; State-dependent delay; Semi-tensor product

https://doi.org/10.1631/FITEE.1900376 **CLC number:** O231

1 Introduction

With the rapid development of gene microarray technology and systems biology, gene regulatory networks (GRNs) have shown great advantages in revealing the interaction mechanism of DNA, mRNA, and protein in cells. Note that Kauffman (1969) initiated the concept of Boolean networks to model GRNs. Using the binary values of "1" and "0" to represent the active and inactive genes respectively, and by describing the interactions between genes by Boolean functions, the Boolean modelling frameworks are easy to simulate and are little computationally taxing. Boolean networks provide a powerful tool for studying GRNs, and abundant excellent results have been established (Akutsu et al., 2007; Ay et al., 2009; Veliz-Cuba and Stigler, 2011). However, the state of a gene is usually not limited to active or inactive in real-world systems. We use a k-valued logical network (logical variables have k different values) as a general network to model GRNs (Chaouiya et al., 2012).

As is well known, because of some time-consuming processes such as DNA translation and RNA translation, time delay is inevitable in the gene regulatory process, which plays a key role in the development of living organisms. Therefore, time delay should be concerned in the logical control networks when modeling the GRNs (Chueh and Lu, 2012; Haider and Pal, 2012; Li FF, 2018). Moreover, the length of the time delay may be related to the current state of the gene. This motivates us to

‡ Corresponding author

* Project supported by the National Natural Science Foundation of China (Nos. 61773371 and 61877036) and the Natural Science Foundation of Shandong Province, China (No. ZR2019MF002)

ORCID: Ya-ting ZHENG, https://orcid.org/0000-0003-2166-5602; Jun-e FENG, https://orcid.org/0000-0003-3881-3042

investigate logical control networks with state-dependent delay, referred to as delayed logical control networks (DLCNs).

Cheng et al. (2012) defined a new mathematical tool, called semi-tensor product (STP), to analyze and control logical control networks. Using the STP method, the dynamics of logical control networks can be converted into a unique bilinear discrete-time system. Then, one can use classic linear control theory to solve a series of problems of logical networks, such as disturbance decoupling (Liu et al., 2017), observability (Fornasini and Valcher, 2013; Yu et al., 2019b), stability (Li YY et al., 2018; Guo et al., 2019; Li H et al., 2019b; Zhu SY et al., 2019a), stabilization (Bof et al., 2015; Lu et al., 2018a; Li XD et al., 2019; Sun et al., 2020), and controllability (Laschov and Margaliot, 2012, 2013; Fornasini and Valcher, 2014; Wu et al., 2019; Zhu QX et al., 2019). The STP method has also been applied to the study of delayed logical (control) networks, and some landmark results have been obtained (Fan et al., 2018; Lu et al., 2018b; Tong et al., 2018; Zhu SY et al., 2018; Li BW et al., 2019; Li H et al., 2019a; Meng et al., 2019; Wang et al., 2019; Yu et al., 2019a).

Note that some virulence genes are expected to avoid synthesis, which may lead to a dangerous situation, such as the deterioration of a disease and the metastasis of a cancer (Lu et al., 2016). In targeted therapy, target gene fragment can be regarded as a constant reference signal. Meanwhile, appropriate medical intervention is necessary, but excessive intervention may cause other side effects for organisms. In logical control networks, the control inputs can be viewed as medicine interventions. Therefore, it is necessary to put some constraints to the states and inputs of the logical networks when modeling the practical GRNs. Moreover, because of the complex internal relationship between states and limitation of measurement equipment, the states are difficult to identify directly. However, it is possible to obtain the structure and function of the logical networks according to the relationship between inputs and outputs (Li HT et al., 2015; Li YY et al., 2019; Zhong et al., 2019; Zhu SY et al., 2019b). In fact, time delay may destroy the controllability of logical networks, driving us to design an admissible controller which steers the outputs of the DLCN to track the given constant signal while avoiding the forbidden states.

2 Preliminaries

In this section, some necessary preparations, including notations, definitions, and properties about STP, are presented. (Cheng et al., 2012).

Notations are as follows:

1. $[m:n] := [m,\ m+1,\ \dots,\ n]$.
2. $\mathcal{D}_k := \{i_1,\ i_2,\ \dots,\ i_k\}$, where $i_1,\ i_2,\ \dots,\ i_k$ are k different integers.
3. $\mathcal{D}_k^n := \underbrace{\mathcal{D}_k \times \mathcal{D}_k \times \ldots \times \mathcal{D}_k}_{n}$.
4. $\mathbf{1}_n^{\mathrm{T}} = [\underbrace{1,\ 1,\ \dots,\ 1}_{n}]$.
5. $\boldsymbol{\delta}_n^i$: the i^{th} column of identity matrix $\boldsymbol{I}_n$.
6. $\Delta_n := \{\boldsymbol{\delta}_n^i : 1 \le i \le n\}$.
7. Matrix $\boldsymbol{L} \in \mathbb{R}^{s\times n}$ is called a logical matrix if $\boldsymbol{L} = [\boldsymbol{\delta}_s^{i_1},\ \boldsymbol{\delta}_s^{i_2},\ \dots,\ \boldsymbol{\delta}_s^{i_n}]$, which can also be briefly expressed as $\boldsymbol{L} = \boldsymbol{\delta}_s[i_1,\ i_2,\ \dots,\ i_n]$, where $i_1,\ i_2,\ \dots,\ i_n \in [1,\ 2,\ \dots,\ s]$.
8. $\mathcal{L}_{s\times n}$: the set of $s \times r$ logical matrices.
9. $(L)_{s,\ n}$: $(s,\ n)^{\text{th}}$ element of the matrix $\boldsymbol{L}$.
10. $\mathrm{Row}_s(\boldsymbol{L})$: s^{th} row of the matrix $\boldsymbol{L}$.
11. $\mathrm{Col}_n(\boldsymbol{L})$: n^{th} column of the matrix $\boldsymbol{L}$.

Definition 1 Let $\boldsymbol{M} \in \mathbb{R}^{m\times n}$ and $\boldsymbol{N} \in \mathbb{R}^{p\times q}$ (Cheng et al., 2012). The STP of $\boldsymbol{M}$ and $\boldsymbol{N}$ is expressed as

$$\boldsymbol{M} \ltimes \boldsymbol{N} = (\boldsymbol{M} \otimes \boldsymbol{I}_{\frac{\alpha}{n}})(\boldsymbol{N} \otimes \boldsymbol{I}_{\frac{\alpha}{p}}), \tag{1}$$

where $\otimes$ represents the Kronecker product and α denotes the least common multiple of n and p.

To facilitate calculation, we identify the k-valued logical variable $i_j \in \mathcal{D}_k$ with a canonical vector $\boldsymbol{\delta}_k^j \in \Delta_k$ $(j = 1,\ 2,\ \dots,\ k)$. By a similar argument in Theorem 2.23 in Li ZQ and Cheng (2010), we have the following Lemmas for the k-valued logical function:

Lemma 1 Any k-valued n-ary logical function $f(\boldsymbol{a}_1,\ \boldsymbol{a}_2,\ \dots,\ \boldsymbol{a}_n)$ with logical variables $\boldsymbol{a}_1, \boldsymbol{a}_2,\ \dots,$ $\boldsymbol{a}_n \in \Delta_k$ can be expressed in a canonical form as (Li ZQ and Cheng, 2010)

$$f(\boldsymbol{a}_1,\ \boldsymbol{a}_2,\ \dots,\ \boldsymbol{a}_n) = \boldsymbol{M}_f \ltimes \boldsymbol{a}_1 \ltimes \boldsymbol{a}_2 \ltimes \ldots \ltimes \boldsymbol{a}_n, \tag{2}$$

where $\boldsymbol{M}_f \in \mathcal{L}_{k\times k^n}$ is the structure matrix of f.

Lemma 2 The STP has the following properties (Cheng et al., 2012):

1. Let $\boldsymbol{X} \in \mathbb{R}^{t\times 1}$ and $\boldsymbol{A} \in \mathbb{R}^{m\times n}$. Then $\boldsymbol{X} \ltimes \boldsymbol{A} = (\boldsymbol{I}_t \otimes \boldsymbol{A}) \ltimes \boldsymbol{X}$.
2. Let $\boldsymbol{X} \in \mathbb{R}^{m\times 1}$ and $\boldsymbol{Y} \in \mathbb{R}^{n\times 1}$. Define a swap matrix $\boldsymbol{W}_{[m,n]} := \boldsymbol{\delta}_{mn}[\boldsymbol{I}_n \otimes \boldsymbol{\delta}_m^1,\ \boldsymbol{I}_n \otimes \boldsymbol{\delta}_m^2,\ \dots,\ \boldsymbol{I}_n \otimes \boldsymbol{\delta}_m^m] \in \mathcal{L}_{mn\times mn}$. Then, $\boldsymbol{Y} \ltimes \boldsymbol{X} = \boldsymbol{W}_{[m,n]} \ltimes \boldsymbol{X} \ltimes \boldsymbol{Y}$.

3. Let $\boldsymbol{A} \in \Delta_m$, $\boldsymbol{B} \in \Delta_n$, and $\boldsymbol{C} \in \Delta_r$ be logical vectors. Define $\boldsymbol{\Psi}[m,\ n,\ r] = \mathbf{1}_m^{\mathrm{T}} \otimes \boldsymbol{I}_n \otimes \mathbf{1}_r^{\mathrm{T}}$.

Then, $\boldsymbol{\Psi}[m,\ n,\ r] \ltimes \boldsymbol{A} \ltimes \boldsymbol{B} \ltimes \boldsymbol{C} = \boldsymbol{B}$.

As a generalization of the conventional matrix product, STP retains all the computational properties of the conventional matrix product. Hence, we omit the symbol "$\ltimes$" if no confusion arises.

3 Main results

In this section, we first convert a DLCN into its algebraic description and then investigate the output tracking of the DLCN with state and input constraints.

3.1 Problem formulation and algebraic transformation

A DLCN with n nodes and p outputs can be described as

$$\begin{cases} a_i(t+1) = f_i(A(t-\mu(A(t))), U(t)), \\ b_j(t) = h_j(A(t-\mu(A(t)))), \end{cases} \tag{3}$$

where $i = 1,\ 2,\ \ldots,\ n$, $j = 1,\ 2,\ \ldots,\ p$, and $a_1,\ a_2,\ \ldots,\ a_n \in \mathcal{D}_k$. $A(t) = (a_1(t),\ a_2(t),\ \ldots,\ a_n(t)) \in \mathcal{D}_k^n$, $U(t) = (u_1(t),\ u_2(t),\ \ldots,\ u_m(t)) \in \mathcal{D}_k^m$, and $B(t) = (b_1(t),\ b_2(t),\ \ldots,\ b_p(t)) \in \mathcal{D}_k^p$ represent the state, control, and output at time t, respectively. $f_i : \mathcal{D}_k^{n+m} \to \mathcal{D}_k$ and $h_j : \mathcal{D}_k^n \to \mathcal{D}_k$ are logical functions. $\mu(A(t))$ denotes the state-dependent delay, which updates values depending on the current state values and a pseudo-logical function $g : \mathcal{D}_k^n \to [0 : \mu^*]$; that is, $\mu(A(t)) = g(A(t)) \in [0 : \mu^*]$.

For a given reference signal $\boldsymbol{B}_{\mathrm{r}} = (b_{\mathrm{r}_1},\ b_{\mathrm{r}_2},\ \ldots,\ b_{\mathrm{r}_p})$, we aim to design an admissible state-feedback controller, expressed as

$$u_i(t) = \varphi_i\big(A(t)\big),\quad i = 1,\ 2,\ \ldots,\ m, \tag{4}$$

where $u_i(t) \in \mathcal{D}_k$ and $\varphi_i : \mathcal{D}_k^n \to \mathcal{D}_k$, under which the output of system (3) is equivalent to B_{r}; that is,

$$\lim_{t\to\infty} B(t) = B_{\mathrm{r}}. \tag{5}$$

To solve the output tracking problem, we first convert system (3) and controller (4) into the equivalent algebraic descriptions.

Identifying the k-valued logical variables as canonical vectors, we define $\boldsymbol{a}(t) = \ltimes_{i=1}^n \boldsymbol{a}_i(t) \in \Delta_{k^n}$, $\boldsymbol{b}(t) = \ltimes_{i=1}^p \boldsymbol{b}_i(t) \in \Delta_{k^p}$, and $\boldsymbol{u}(t) = \ltimes_{i=1}^m \boldsymbol{u}_i(t) \in \Delta_{k^m}$. Following Lemma 1, it is easy to calculate that

$$\begin{cases} \boldsymbol{a}(t+1) = \boldsymbol{F}\boldsymbol{u}(t)\boldsymbol{a}(t-\mu(\boldsymbol{a}(t))), \\ \boldsymbol{b}(t) = \boldsymbol{H}\boldsymbol{a}(t-\mu(\boldsymbol{a}(t))), \end{cases} \tag{6}$$

$$\boldsymbol{u}(t) = \boldsymbol{\Phi}\boldsymbol{a}(t), \tag{7}$$

$$\mu(\boldsymbol{a}(t)) = \boldsymbol{G}\boldsymbol{a}(t), \tag{8}$$

where $\boldsymbol{F} \in \mathcal{L}_{k^n \times k^{m+n}}$, $\boldsymbol{H} \in \mathcal{L}_{k^p \times k^n}$, $\boldsymbol{\Phi} \in \mathcal{L}_{k^m \times k^n}$, and $\boldsymbol{G} \in \mathbb{R}^{1\times k^n}$.

According to the values of the state-dependent delay, we classify the state values into several different sets as follows:

$$\varUpsilon_\tau = \{\boldsymbol{a}(t) | \mu(\boldsymbol{a}(t)) = \tau,\ \ \tau \in [0 : \mu^*]\}. \tag{9}$$

Specifically, we define $\varUpsilon_\tau = \varnothing$ if there is no state value that satisfies $\mu(\boldsymbol{a}(t)) = \tau$. Clearly, $\varUpsilon_i \cap \varUpsilon_j = \varnothing$ $(i \neq j)$, and $\cup_{i=0}^{\mu^*} \varUpsilon_i = \Delta_{k^n}$.

Define a series of row vectors $\boldsymbol{M}_i \in \mathbb{R}^{1\times k^n}$ $(i = 0,\ 1,\ \ldots,\ \mu^*)$ as

$$\mathrm{Col}_j(\boldsymbol{M}_i) = \begin{cases} 1,\ \text{if } \boldsymbol{\delta}_{k^n}^j \in \varUpsilon_i, \\ 0,\ \text{otherwise}, \end{cases} \tag{10}$$

where $j = 1,\ 2,\ \ldots,\ k^n$. Then

$$\boldsymbol{M}_i \boldsymbol{a}(t) = \begin{cases} 1,\ \text{if } \boldsymbol{a}(t) \in \varUpsilon_i, \\ 0,\ \text{otherwise}. \end{cases} \tag{11}$$

Now, we remove the coupling from the first equation of system (6). Suppose that $\mu(\boldsymbol{a}(t)) = \tau \in [0 : \mu^*]$ at time t, and then $\boldsymbol{a}(t) \in \varUpsilon_\tau$. Let $\boldsymbol{x}(t) = \ltimes_{i=t-\mu^*}^t \boldsymbol{a}(i) \in \Delta_{k^{(\mu^*+1)n}}$. Then the straightforward calculation shows that

$$\begin{aligned} \boldsymbol{a}(t+1) =& \boldsymbol{F}\boldsymbol{u}(t)\boldsymbol{a}(t-\tau) \\ =& \boldsymbol{M}_\tau \boldsymbol{a}(t) \otimes (\boldsymbol{F}\boldsymbol{u}(t)\boldsymbol{a}(t-\tau)) \\ =& (\boldsymbol{M}_\tau \otimes \boldsymbol{F})\boldsymbol{W}_{(k^n,\ k^{m+n})}\boldsymbol{\Psi}_{[k^{(\mu^*-\tau)n},\ k^n,\ k^{\tau n}]} \\ & \ltimes \boldsymbol{x}(t)\boldsymbol{a}(t)\boldsymbol{u}(t) \\ =& (\boldsymbol{M}_\tau \otimes \boldsymbol{F})\boldsymbol{W}_{(k^n,\ k^{m+n})}\boldsymbol{\Psi}_{[k^{(\mu^*-\tau)n},\ k^n,\ k^{\tau n}]} \\ & \ltimes (\boldsymbol{I}_{k^{\mu^* n}} \otimes \boldsymbol{M}_{r,\ k^n})\boldsymbol{W}_{[k^m,\ k^{(\mu^*+1)n}]}\boldsymbol{u}(t)\boldsymbol{x}(t) \\ :=& \boldsymbol{F}_\tau \boldsymbol{u}(t)\boldsymbol{x}(t). \end{aligned} \tag{12}$$

System (12) can be rewritten as follows:

$$\begin{aligned} \boldsymbol{a}(t+1) &= \boldsymbol{F}_0\boldsymbol{u}(t)\boldsymbol{x}(t) + \boldsymbol{F}_1\boldsymbol{u}(t)\boldsymbol{x}(t) + \ldots \\ &\quad + \boldsymbol{F}_{\mu^*}\boldsymbol{u}(t)\boldsymbol{x}(t) \\ &:= \hat{\boldsymbol{F}}\boldsymbol{u}(t)\boldsymbol{x}(t), \end{aligned} \tag{13}$$

where $\hat{\boldsymbol{F}} = \boldsymbol{F}_0 + \boldsymbol{F}_1 + \ldots + \boldsymbol{F}_{\mu^*} \in \mathcal{L}_{k^n \times k^{[m+(\mu^*+1)n]}}$.

Via a similar conversion process, the output and controller can be rewritten as follows:

$$\begin{cases}\boldsymbol{b}(t) = \hat{\boldsymbol{H}}\boldsymbol{x}(t),\\ \boldsymbol{u}(t) = \hat{\boldsymbol{\Phi}}\boldsymbol{x}(t),\end{cases} \tag{14}$$

where $\hat{\boldsymbol{H}} \in \mathcal{L}_{k^p \times k^{(\mu^*+1)n}}$ and $\hat{\boldsymbol{\Phi}} \in \mathcal{L}_{k^m \times k^{(\mu^*+1)n}}$.

We augment system (12) into the following system:

$$\begin{aligned}\boldsymbol{x}(t+1) =&\boldsymbol{a}(t-\mu^*+1)\boldsymbol{a}(t-\mu^*+2)\ldots\boldsymbol{a}(t+1)\\ =&\boldsymbol{\Psi}_{(k^n,\ k^{\mu^* n},\ 1)}\boldsymbol{a}(t-\mu^*)\ldots\boldsymbol{a}(t)\hat{\boldsymbol{F}}\boldsymbol{u}(t)\boldsymbol{x}(t)\\ =&\boldsymbol{\Psi}_{(k^n,\ k^{\mu^* n},\ 1)}\boldsymbol{W}_{[k^n,\ k^{(\mu^*+1)n}]}\\ &\ltimes \hat{\boldsymbol{F}}(\boldsymbol{I}_{k^m} \otimes \boldsymbol{M}_{[r,\ k^{(\mu^*+1)n}]})\boldsymbol{u}(t)\boldsymbol{x}(t)\\ :=&\boldsymbol{\Theta}\boldsymbol{u}(t)\boldsymbol{x}(t),\end{aligned} \tag{15}$$

where $\boldsymbol{\Theta} \in \mathcal{L}_{k^{(\mu^*+1)n} \times k^{m+(\mu^*+1)n}}$.

From the analysis above, we can see that any state sequence $(\boldsymbol{a}(t-\mu^*),\ \boldsymbol{a}(t-\mu^*+1),\ \ldots,\ \boldsymbol{a}(t))$ of system (13) corresponds to a unique state $\boldsymbol{x}(t) = \ltimes_{i=t-\mu^*}^{t}\boldsymbol{a}(i)$ of system (15). Moreover, any state $\boldsymbol{x}(t)$ of system (15) can be decomposed into a unique state sequence $(\boldsymbol{a}(t-\mu^*),\ \boldsymbol{a}(t-\mu^*+1),\ \ldots,\ \boldsymbol{a}(t))$ of system (13). Therefore, system (13) is equivalent to system (15), and we can focus on the augmented system (15) to investigate the output tracking problem of the original system (3).

3.2 Set stabilization of DLCNs with state and control constraints

To avoid some undesirable states which stand for virulence genes of the GRNs, we constrain the state values of system (13) on a nonempty set Γ_a, which is defined as

$$\Gamma_a = \{\boldsymbol{\delta}_{k^n}^{\gamma_1},\ \boldsymbol{\delta}_{k^n}^{\gamma_2},\ \ldots,\ \boldsymbol{\delta}_{k^n}^{\gamma_\zeta}\} \subseteq \Delta_{k^n}, \tag{16}$$

where $\gamma_1 < \gamma_2 < \ldots < \gamma_\zeta$.

By multiplying any μ^*+1 value of Γ_a, we construct the following set:

$$\Gamma_x = \{\boldsymbol{\delta}_{k^{(\mu^*+1)n}}^{\xi_1},\ \boldsymbol{\delta}_{k^{(\mu^*+1)n}}^{\xi_2},\ \ldots,\ \boldsymbol{\delta}_{k^{(\mu^*+1)n}}^{\xi_{\zeta^{\mu^*+1}}}\}, \tag{17}$$

where $\xi_1 < \xi_2 < \ldots < \xi_{\zeta^{\mu^*+1}}$. Then, the state values of augmented system (15) belong to Γ_x. It should be noted that if $\boldsymbol{x}(t+1) = \boldsymbol{\Theta}\boldsymbol{u}(t)\boldsymbol{x}(t) \notin \Gamma_a$, then $\boldsymbol{x}(t+1)$ is a forbidden value. Set $C_x = \{\xi_1,\ \xi_2,\ \ldots,\ \xi_{\zeta^{\mu^*+1}}\}$.

From a practical point of view, we constrain the control values on the nonempty set Γc, which is defined as

$$\Gamma_{\mathrm{c}} = \{\boldsymbol{\delta}_{k^m}^{\nu_1},\ \boldsymbol{\delta}_{k^m}^{\nu_2},\ \ldots,\ \boldsymbol{\delta}_{k^m}^{\nu_s}\}. \tag{18}$$

In the rest of this work, system (13) with state constraint (16) and control constraint (18) is simply referred to as constrained system (13). System (15) with state constraint (17) and control constraint (18) is simply referred to as constrained system (15) for narrative purposes.

Before presenting the results, we introduce two symbols, which will be used later. Given a control sequence $u = \{\boldsymbol{u}(0),\ \boldsymbol{u}(1),\ \ldots,\ \boldsymbol{u}(t-1)\}$, it denotes that the state and output constraints of system (3) start from initial state $\boldsymbol{x}_0 := \boldsymbol{a}(-\mu^*)\boldsymbol{a}(-\mu^*+1)\ldots\boldsymbol{a}(0)$ at time t by $\boldsymbol{a}(t;\boldsymbol{x}_0,\ u)$ and $\boldsymbol{b}(t;\boldsymbol{x}_0,\ u)$, respectively. Set $\boldsymbol{x}(t;\boldsymbol{x}_0,\ u) := \boldsymbol{a}(t-\mu^*;\boldsymbol{x}_0,\ u)\boldsymbol{a}(t-\mu^*+1;\boldsymbol{x}_0,\ u)\ldots\boldsymbol{a}(t;\boldsymbol{x}_0,\ u)$.

Definition 2 Constrained system (13) is Γ_a-stabilizable, if there exists an integer $T \in \mathbb{N}^+$ and a state-feedback control sequence $u = \{\boldsymbol{u}(0),\ \boldsymbol{u}(1),\ \ldots,\ \boldsymbol{u}(t-1)\}$ with $\boldsymbol{u}(i) \in \Gamma_{\mathrm{c}}$ $(i = 0,\ 1,\ \ldots,\ t-1)$, such that

$$\boldsymbol{a}(t;\boldsymbol{x}_0,\ u) \in \Gamma_a,\ \forall\ t \geq T, \tag{19}$$

where $\boldsymbol{x}_0 \in \Gamma_x$ is an arbitrary initial state.

Similarly, the definition of set stabilization for system (15) can be drawn as follows:

Definition 3 Constrained system (15) is Γ_x-stabilizable, for any initial state $\boldsymbol{x}_0 \in \Gamma_x$, if there exists an integer $T \in \mathbb{N}^+$ and a state-feedback control sequence $u = \{\boldsymbol{u}(0),\ \boldsymbol{u}(1),\ \ldots,\ \boldsymbol{u}(t-1)\}$ with $\boldsymbol{u}(i) \in \Gamma_c$ $(i = 0,\ 1,\ \ldots,\ t-1)$, such that

$$\boldsymbol{x}(t;\boldsymbol{x}_0,\ u) \in \Gamma_x,\ \forall\ t \geq T. \tag{20}$$

Lemma 3 Constrained system (13) is Γ_a-stabilizable via $\boldsymbol{u}(t) = \hat{\boldsymbol{H}}\boldsymbol{x}(t)$, if and only if constrained system (15) is Γ_x-stabilizable via $\boldsymbol{u}(t) = \hat{\boldsymbol{H}}\boldsymbol{x}(t)$.

Set $\boldsymbol{b}_{\mathrm{r}} = \ltimes_{i=1}^{p}\boldsymbol{b}_{\mathrm{r}_i} := \boldsymbol{\delta}_{k^p}^{\epsilon}$. According to the reference signal $\boldsymbol{\delta}_{k^p}^{\epsilon}$ and output matrix $\hat{\boldsymbol{H}}$, we denote a target set Ω, which contains all the states of constrained system (15) whose outputs are equivalent to $\boldsymbol{\delta}_{k^p}^{\epsilon}$, as follows:

$$\begin{aligned}\Omega =&\{\boldsymbol{\delta}_{k^{(\mu^*+1)n}}^{\xi}|\mathrm{Col}_\xi(\hat{\boldsymbol{H}}) = \boldsymbol{\delta}_{k^p}^{\epsilon},\ \xi \in C_x\}\\ :=&\{\boldsymbol{\delta}_{k^{(\mu^*+1)n}}^{\beta_1},\ \boldsymbol{\delta}_{k^{(\mu^*+1)n}}^{\beta_2},\ \ldots,\ \boldsymbol{\delta}_{k^{(\mu^*+1)n}}^{\beta_d}\},\end{aligned} \tag{21}$$

where $\beta_1 < \beta_2 < \ldots < \beta_d$.

Following Definition 3, we derive the conclusion as follows:

Theorem 1 The outputs of system (3) with multi-constraint track the reference signal $\boldsymbol{b}_{\mathrm{r}}$ via state-feedback control, if and only if constrained system (15) is Ω-stabilizable.

Proof Suppose that the outputs of system (3) with state constraint (16) and control constraint (18) track the reference signal $\boldsymbol{b}_{\mathrm{r}}$. Because the states of system (3) have at most ζ^{μ^*+1} different values, there exists a positive integer $T \le \zeta^{\mu^*+1}$ and an admissible state-feedback control sequence u, such that $\boldsymbol{b}(t;\boldsymbol{x}_0,\ u) = \boldsymbol{b}_{\mathrm{r}}$ holds for any integer $t \ge T$ and any initial state $\boldsymbol{x}_0 \in \Gamma_x$. Then, we can see that $\boldsymbol{x}(t;\boldsymbol{x}_0,\ u) \in \Omega$ holds, which combined with Definition 3 shows that constrained system (15) is Ω-stabilizable.

Conversely, suppose that constrained system (15) is Ω-stabilizable. Following Definition 3, there exists an admissible state-feedback control sequence u and a positive integer $T \le \zeta^{\mu^*+1}$, such that $\boldsymbol{x}(t;\boldsymbol{x}_0,\ u) \in \Omega$ holds for any integer $t \ge T$ and any initial state $\boldsymbol{x}_0 \in \Gamma_x$. Furthermore, the composition of set Ω implies that $\hat{\boldsymbol{H}}\boldsymbol{x}(t;\boldsymbol{x}_0,\ u) = \boldsymbol{b}_{\mathrm{r}}$. Thus, the outputs of system (3) with multi-constraint track the reference signal $\boldsymbol{b}_{\mathrm{r}}$.

3.3 Output tracking of DLCNs with state and control constraints

Considering system (15) and control constraint Γ_{c}, split $\boldsymbol{\Theta}$ into k^m equal blocks and then sum up these block matrices with indices $\nu_1, \nu_2, \ldots, \nu_s$ as follows:

$$\boldsymbol{\Xi} = \Sigma_{i=1}^{s} \boldsymbol{\Theta}_{\nu_i}, \tag{22}$$

where $\boldsymbol{\Theta}_{\nu_i} = \boldsymbol{\Theta}\boldsymbol{\delta}_{k^m}^{\nu_i}$ $(i = 1,\ 2,\ \ldots,\ s)$.

Lemma 4 As a special deformation of the state transition matrix, $\boldsymbol{\Xi}$ effectively reflects the reachability between any two states of system (15). For example, $(\boldsymbol{\Xi}^s)_{p,\ q} > 0$ denotes that there exists an admissible state-feedback control sequence driving $\boldsymbol{\delta}_{k^{(\mu^*+1)n}}^{q}$ to $\boldsymbol{\delta}_{k^{(\mu^*+1)n}}^{p}$ at the s^{th} step.

In the following, we simplify the matrix $\boldsymbol{\Xi}$ to determine the kernel-attractor set Ω.

Set

$$\boldsymbol{\Xi}_{\Omega} = \boldsymbol{E}^{\mathrm{T}}\boldsymbol{\Xi}\boldsymbol{E}, \tag{23}$$

where $\boldsymbol{E} = [\boldsymbol{\delta}_{k^{(\mu^*+1)n}}^{\beta_1},\ \boldsymbol{\delta}_{k^{(\mu^*+1)n}}^{\beta_2},\ \ldots,\ \boldsymbol{\delta}_{k^{(\mu^*+1)n}}^{\beta_d}]$.

Then, the kernel-attractor set is constructed as follows:

$$\begin{aligned} N =& \left\{ \boldsymbol{\delta}_{k^{(\mu^*+1)n}}^{\beta_j} \mid \Big(\sum_{s=1}^{d} (\boldsymbol{\Xi}_{\Omega})^s\Big)_{j,\ j} > 0 \right\} \\ :=& \{\boldsymbol{\delta}_{k^{(\mu^*+1)n}}^{\beta_1'},\ \boldsymbol{\delta}_{k^{(\mu^*+1)n}}^{\beta_2'},\ \ldots,\ \boldsymbol{\delta}_{k^{(\mu^*+1)n}}^{\beta_z'}\}, \end{aligned} \tag{24}$$

where $N \subseteq \Omega$ contains all the kernel attractors of Ω. Set $C_N = \{\beta_1',\ \beta_2',\ \ldots,\ \beta_z'\}$.

We define a new matrix $\boldsymbol{Q}_N$ as follows:

$$\boldsymbol{Q}_N = \boldsymbol{R}^{\mathrm{T}}\boldsymbol{\Xi}\boldsymbol{R}, \tag{25}$$

where $\operatorname{rank}(\boldsymbol{R}) = \zeta^{\mu^*+1} - z + 1$ and

$$\begin{cases} \operatorname{Col}_1(\boldsymbol{R}) = \sum_{j=1}^{z} \boldsymbol{\delta}_{k^{(\mu^*+1)n}}^{\beta_j'}, \\ \operatorname{Col}_i(\boldsymbol{R}) = \boldsymbol{\delta}_{k^{(\mu^*+1)n}}^{\xi} \in \Gamma_x \setminus N,\ i \in [2 : \zeta^{\mu^*+1} - z]. \end{cases} \tag{26}$$

Theorem 2 The outputs of system (3) with multi-constraint track the reference signal $\boldsymbol{b}_{\mathrm{r}}$ via state-feedback control, if and only if there exists an integer $\eta \in [1 : \zeta^{\mu^*+1} - z]$ and a kernel-attractor set $N \subseteq \Omega$, such that

$$\operatorname{Row}_1(\boldsymbol{Q}_N^{\eta}) > 0. \tag{27}$$

Proof Suppose that the outputs of system (3) with state constraint (16) and control constraint (18) track the reference signal $\boldsymbol{b}_{\mathrm{r}}$ via $\boldsymbol{u} = \hat{\boldsymbol{\Phi}}\boldsymbol{x}(t)$ within a finite time, which together with Theorem 1 reflects that constrained system (15) is Ω-stabilizable. Then, we can obtain the following closed-loop system:

$$\boldsymbol{x}(t+1) = \hat{\boldsymbol{\Theta}}\boldsymbol{x}(t), \tag{28}$$

where $\hat{\boldsymbol{\Theta}} = \boldsymbol{\Theta}\hat{\boldsymbol{\Phi}}\boldsymbol{M}_{r,\ k^{(\mu^*+1)n}}$. Determine the kernel-attractor set $N := \{\boldsymbol{\delta}_{k^{(\mu^*+1)n}}^{\beta_1'},\ \boldsymbol{\delta}_{k^{(\mu^*+1)n}}^{\beta_2'},\ \ldots,\ \boldsymbol{\delta}_{k^{(\mu^*+1)n}}^{\beta_z'}\} \subseteq \Omega$ from Eq. (24). Denote the state transition period of system (28) by η. Then, for any initial state $\boldsymbol{x}_0 \in \Gamma_x$, there exists an admissible control sequence $u = \{\boldsymbol{u}(0),\ \boldsymbol{u}(1),\ \ldots,\ \boldsymbol{u}(\eta-1)\}$, such that $\boldsymbol{x}(\eta;\boldsymbol{x}_0,\ u) \in N$. Besides kernel attractors, there are $\zeta^{\mu^*+1} - z$ different states; hence, $\eta \in [1 : \zeta^{\mu^*+1} - z]$. The matrix $\boldsymbol{R}$ defined in Eq. (26) is used to pick up the useful rows and columns of matrix $\boldsymbol{\Xi}$ with indices $\{\beta_1',\ \beta_2',\ \ldots,\ \beta_z'\}$, which combined with Lemma 4 shows that Eq. (27) holds.

Conversely, assume that Eq. (27) holds. It is easy to derive that $\operatorname{Row}_1(\boldsymbol{Q}_N^{s}) > 0$ holds for any integer $s > \eta$, reflecting that for any state $\boldsymbol{x}_0 \in \Gamma_x$, there exists an admissible control sequence u such that $\boldsymbol{x}(s;\boldsymbol{x}_0,\ u)\in N\subseteq\Omega$ holds for any $s\geq\eta$. By Definition 3, we can see that constrained system (15) is Ω-stabilizable. Moreover, following Theorem 1, the outputs of system (3) with multi-constraint track the reference signal $\boldsymbol{b}_{\mathrm{r}}$.

Suppose that Eq. (27) holds. We consider to design the admissible state-feedback controller to enable the outputs of system (3) with multi-constraint to track the reference signal $\boldsymbol{b}_{\mathrm{r}}$. First, the reachable sets of the kernel-attractor set N are constructed as follows:

$$\begin{aligned}R_1(N)=&\{\boldsymbol{\delta}^i_{k^{(\mu^*+1)n}}|\boldsymbol{\Xi}_{\beta'_i,\ i}>0,\\&\forall\beta'_i\in C_N, i\in C_x\}\backslash N,\end{aligned}\tag{29}$$

$$\begin{aligned}R_s(N)=&\{\boldsymbol{\delta}^j_{k^{(\mu^*+1)n}}|\boldsymbol{\Xi}^s_{\beta'_i,j}>0,\ \forall\beta'_i\in C_N,\\&j\in C_x\}\backslash R_{s-1},\ s=2,\ 3,\ \ldots,\ \zeta^{\mu^*+1}-z.\end{aligned}\tag{30}$$

Clearly, $R_i(N)\cap R_j(N)=\varnothing$ $(\forall i\neq j)$ and $\cup_{i=1}^{\zeta^{\mu^*+1}-z}R_i(N)\cup N=\Gamma_x$. Then, we focus on designing a state-feedback controller $\boldsymbol{u}(t)=\boldsymbol{\delta}^{p_i}_{k^m}\in\Gamma_{\mathrm{c}}$ to achieve the following objectives:

1. For any state $\boldsymbol{x}(t)=\boldsymbol{\delta}^i_{k^{(\mu^*+1)n}}\in N\cup R_1$, we achieve $\boldsymbol{x}(t+1)=\boldsymbol{\Theta}_{p_j}\boldsymbol{\delta}^i_{k^{(\mu^*+1)n}}\in N$.

2. For any state $\boldsymbol{x}(t)=\boldsymbol{\delta}^i_{k^{(\mu^*+1)n}}\in R_s$, we achieve $\boldsymbol{x}(t+1)=\boldsymbol{\Theta}_{p_j}\boldsymbol{\delta}^i_{k^{(\mu^*+1)n}}\in R_{s-1}$, $2\leq s\leq\zeta^{\mu^*+1}-z$.

Based on the analysis above, the state-feedback controller can be designed as follows:

$$\boldsymbol{u}(t)=\boldsymbol{\delta}_{k^m}(p_1,\ p_2,\ \ldots,\ p_{k^{(\mu^*+1)n}})\boldsymbol{x}(t).\tag{31}$$

4 An illustrative example

In this section, we give an illustrative example to show the effectiveness of the results.

Example 1 Consider the following DLCN with $\boldsymbol{G}=[1,\ 1,\ 0,\ 1,\ 0,\ 1,\ 1,\ 1]$, which models a biological network. Refer to Veliz-Cuba and Stigler (2011) for details.

$$\begin{cases}a_1(t+1)=\neg u_1(t)\wedge(a_2(t-\mu(A(t)))\\ \qquad\qquad\vee a_3(t-\mu(A(t)))),\\ a_2(t+1)=\neg u_1(t)\wedge u_2(t)\wedge a_1(t-\mu(A(t))),\\ a_3(t+1)=\neg u_1(t)\wedge(u_2(t)\vee(u_3(t)\\ \qquad\qquad\wedge a_1(t-\mu(A(t))))),\end{cases}\tag{32}$$

where $x_i\in\mathcal{D}_2$ $(i=1,\ 2,\ 3)$ and $A(t)=(a_1(t),\ a_2(t),\ a_3(t))$.

In this example, to observe the effect of delay on the state, we specify that the outputs depict the state values with state-dependent delay; that is, $b_i(t)=a_i(t-\mu(A(t)))$, where $i=1,\ 2,\ 3$.

Converting the logical variables into the canonical vector form, we define $\boldsymbol{a}(t)=\ltimes^3_{i=1}\boldsymbol{a}_i(t)\in\Delta_8$, $\boldsymbol{b}(t)=\ltimes^3_{i=1}\boldsymbol{b}_i(t)\in\Delta_8$, $\boldsymbol{u}(t)=\ltimes^3_{i=1}\boldsymbol{u}_i(t)\in\Delta_8$, and $\boldsymbol{x}(t)=\boldsymbol{a}(t-1)\boldsymbol{a}(t)\in\Delta_{64}$.

Suppose that the state constraint is $\Gamma_a=\{\boldsymbol{\delta}^1_8,\ \boldsymbol{\delta}^2_8,\ \boldsymbol{\delta}^4_8,\ \boldsymbol{\delta}^6_8\}$ and the control constraint is $\Gamma_{\mathrm{c}}=\{\boldsymbol{\delta}^5_8,\ \boldsymbol{\delta}^6_8,\ \boldsymbol{\delta}^7_8\}$. We aim to design a state-feedback controller under which the outputs of system (32) track the given reference signal $\boldsymbol{b}_{\mathrm{r}}=\boldsymbol{\delta}^1_8$. Following Eqs. (9) and (10), we classify the states into the following two sets:

1. $\Upsilon_0=\{\boldsymbol{\delta}^3_8,\ \boldsymbol{\delta}^5_8\}$ and $\boldsymbol{M}_0=[0,\ 0,\ 1,\ 0,\ 1,\ 0,\ 0,\ 0]$.

2. $\Upsilon_1=\{\boldsymbol{\delta}^1_8,\ \boldsymbol{\delta}^2_8,\ \boldsymbol{\delta}^4_8,\ \boldsymbol{\delta}^6_8,\ \boldsymbol{\delta}^7_8,\ \boldsymbol{\delta}^8_8\}$ and $\boldsymbol{M}_1=[1,\ 1,\ 0,\ 1,\ 0,\ 1,\ 1,\ 1]$.

According to Eqs. (12)–(14), system (32) can be converted into the following form:

$$\begin{cases}\boldsymbol{a}(t+1)=\hat{\boldsymbol{F}}\boldsymbol{u}(t)\boldsymbol{x}(t),\\ \boldsymbol{b}(t)=\hat{\boldsymbol{H}}\boldsymbol{x}(t),\end{cases}\tag{33}$$

where

$$\begin{aligned}\hat{\boldsymbol{H}}=\boldsymbol{\delta}_8[&1,1,3,1,5,1,1,1,2,2,3,2,5,2,2,2,\\&3,3,3,3,5,3,3,3,4,4,3,4,5,4,4,4,\\&\cdots\\&5,5,3,5,5,5,5,5,6,6,3,6,5,6,6,6,\\&7,7,3,7,5,7,7,7,8,8,3,8,5,8,8,8]\\&\in\mathcal{L}_{8\times64}.\end{aligned}\tag{34}$$

Following Eq. (15), we convert the first equation of system (33) into the following augmented system:

$$\boldsymbol{x}(t+1)=\boldsymbol{\Theta}\boldsymbol{u}(t)\boldsymbol{x}(t),\tag{35}$$

where

$$\begin{aligned}\boldsymbol{\Theta}=\boldsymbol{\delta}_{64}[&4,12,20,28,36,44,52,60,8,16,\\&36,48,56,64,4,12,20,28,36,44,\\&\cdots\\&36,48,56,64,4,12,20,28,36,44,\\&52,60,8,16,20,32,36,48,56,64]\\&\in\mathcal{L}_{64\times512}.\end{aligned}\tag{36}$$

Multiplying any two states in Γ_a, we obtain the following augmented state constraint: $\Gamma_x=$

$\{\boldsymbol{\delta}_{64}^{1},\ \boldsymbol{\delta}_{64}^{2},\ \boldsymbol{\delta}_{64}^{3},\ \boldsymbol{\delta}_{64}^{8},\ \boldsymbol{\delta}_{64}^{9},\ \boldsymbol{\delta}_{64}^{10},\ \boldsymbol{\delta}_{64}^{11},\ \boldsymbol{\delta}_{64}^{16},\ \boldsymbol{\delta}_{64}^{17},\ \boldsymbol{\delta}_{64}^{18},\ \boldsymbol{\delta}_{64}^{19},\ \boldsymbol{\delta}_{64}^{24},\ \boldsymbol{\delta}_{64}^{49},\ \boldsymbol{\delta}_{64}^{50},\ \boldsymbol{\delta}_{64}^{51},\ \boldsymbol{\delta}_{64}^{56},\ \boldsymbol{\delta}_{64}^{57},\ \boldsymbol{\delta}_{64}^{58},\ \boldsymbol{\delta}_{64}^{59},\ \boldsymbol{\delta}_{64}^{64}\}$.

Based on matrix $\hat{\boldsymbol{H}}$, we compute that $\varOmega = \{\boldsymbol{\delta}_{64}^{1},\ \boldsymbol{\delta}_{64}^{2},\ \boldsymbol{\delta}_{64}^{8}\}$.

Following Eqs. (23) and (27), $\boldsymbol{\varXi}$ and $\boldsymbol{Q}$ are easily calculated. Let $N = \{\boldsymbol{\delta}_{64}^{1}\}$. Then, we derive that $\mathrm{Row}_1(\boldsymbol{Q}_N^{16}) > 0$. By Theorem 2, we can see that the outputs of system (32) track the reference signal $\boldsymbol{b}_\mathrm{r}$ via the following state-feedback controller:

$$\boldsymbol{u}(t) = \boldsymbol{\delta}_8[i_1,\ i_2,\ \ldots,\ i_{64}]\boldsymbol{x}(t), \tag{37}$$

where

$$\begin{cases} i_j = 5,\ j = 1, 2, 3, 9, 10, 11, 16, 17, 18, 19, 50, \\ \qquad\qquad 51,\ 56,\ 57,\ 58,\ 59,\ 63,\ 64, \\ i_j = 7,\ j = 7,\ 8, 15,\ 23,\ 24,\ ...,\ 48. \end{cases} \tag{38}$$

5 Conclusions

In summary, using the STP method we have converted the dynamics of the DLCN into a bilinear discrete-time system and constructed an equivalent augmented system. Based on the augmented system, we have transformed the output tracking problem into a set stabilization problem. Considering the state and control constraints, we have modified the state transition matrix and presented some necessary and sufficient conditions for output tracking of the system. In addition, we have designed the state-feedback controller to drive the outputs of the constrained DLCN to track the reference signal. We have given an illustrative example to show that the new results are effective.

Contributors

Ya-ting ZHENG drafted the manuscript. Jun-e FENG revised and finalized the manuscript.

Compliance with ethics guidelines

Ya-ting ZHENG and Jun-e FENG declare that they have no conflict of interest.

References

Akutsu T, Hayashida M, Ching WK, et al., 2007. Control of Boolean networks: hardness results and algorithms for tree structured networks. *J Theor Biol*, 244(4):670-679. https://doi.org/10.1016/j.jtbi.2006.09.023

Ay F, Xu F, Kahveci T, 2009. Scalable steady state analysis of Boolean biological regulatory networks. *PLoS ONE*, 4(12):e7992. https://doi.org/10.1371/journal.pone.0007992

Bof N, Fornasini E, Valcher ME, 2015. Output feedback stabilization of Boolean control networks. *Automatica*, 57:21-28. https://doi.org/10.1016/j.automatica.2015.03.032

Chaouiya C, Naldi A, Thieffry D, 2012. Logical modelling of gene regulatory networks with GINsim. In: van Helden J, Toussaint A, Thieffry D (Eds.), Bacterial Molecular Networks. Springer, New York, p.463-479. https://doi.org/10.1007/978-1-61779-361-5_23

Cheng D, Qi H, Zhao Y, 2012. An Introduction to Semi-tensor Product of Matrices and its Applications. World Scientific, Singapore.

Chueh TH, Lu HHS, 2012. Inference of biological pathway from gene expression profiles by time delay Boolean networks. *PLoS ONE*, 7(8):e42095. https://doi.org/10.1371/journal.pone.0042095

Fan HB, Feng JE, Meng M, et al., 2018. General decomposition of fuzzy relations: semi-tensor product approach. *Fuzzy Set Syst*, p.1-16. https://doi.org/10.1016/j.fss.2018.12.012

Fornasini E, Valcher ME, 2013. Observability, reconstructibility and state observers of Boolean control networks. *IEEE Trans Autom Contr*, 58(6):1390-1401. https://doi.org/10.1109/TAC.2012.2231592

Fornasini E, Valcher ME, 2014. Optimal control of Boolean control networks. *IEEE Trans Autom Contr*, 59(5):1258-1270. https://doi.org/10.1109/TAC.2013.2294821

Guo YQ, Zhou RP, Wu YH, et al., 2019. Stability and set stability in distribution of probabilistic Boolean networks. *IEEE Trans Autom Contr*, 64(2):736-742. https://doi.org/10.1109/TAC.2018.2833170

Haider S, Pal R, 2012. Boolean network inference from time series data incorporating prior biological knowledge. *BMC Genom*, 13:S9. https://doi.org/10.1186/1471-2164-13-S6-S9

Kauffman S, 1969. Metabolic stability and epigenesis in randomly constructed genetic nets. *J Theor Biol*, 22(3):437-467. https://doi.org/10.1016/0022-5193(69)90015-0

Laschov D, Margaliot M, 2012. Controllability of Boolean control networks via the Perro–Frobenius theory. *Automatica*, 48(6):1218-1223. https://doi.org/10.1016/j.automatica.2012.03.022

Laschov D, Margaliot M, 2013. Minimum-time control of Boolean networks. *SIAM J Contr Optim*, 51(4):2869-2892. https://doi.org/10.1137/110844660

Li BW, Lou JG, Liu Y, et al., 2019. Robust invariant set analysis of Boolean networks. *Complexity*, 2019:2731395. https://doi.org/10.1155/2019/2731395

Li FF, 2018. Stability of Boolean networks with delays using pinning control. *IEEE Trans Contr Netw Syst*, 5(1):179-185. https://doi.org/10.1109/TCNS.2016.2585861

Li H, Zheng Y, Alsaadi F, 2019a. Algebraic formulation and topological structure of Boolean networks with state-dependent delay. *J Comput Appl Math*, 350:87-97. https://doi.org/10.1016/j.cam.2018.10.003

Li H, Xu X, Ding X, 2019b. Finite-time stability analysis of stochastic switched Boolean networks with impulsive effect. *Appl Math Comput*, 347:557-565. https://doi.org/10.1016/j.amc.2018.11.018

Li HT, Wang YZ, Xie LH, 2015. Output tracking control of Boolean control networks via state feedback: constant reference signal case. *Automatica*, 59:54-59. https://doi.org/10.1016/j.automatica.2015.06.004

Li XD, Li HT, Zhao GD, 2019. Function perturbation impact on feedback stabilization of Boolean control networks. *IEEE Trans Neur Netw Learn Syst*, 30(8):2548-2554. https://doi.org/10.1109/TNNLS.2018.2881168

Li YY, Li BW, Liu Y, et al., 2018. Set stability and set stabilization of switched Boolean networks with state-based switching. *IEEE Access*, 6:35624-35630. https://doi.org/10.1109/ACCESS.2018.2851391

Li YY, Liu RJ, Lou JG, et al., 2019. Output tracking of Boolean control networks driven by constant reference signal. *IEEE Access*, 7:112572-112577. https://doi.org/10.1109/ACCESS.2019.2934740

Li ZQ, Cheng DZ, 2010. Algebraic approach to dynamics of multivalued networks. *Int J Bifurc Chaos*, 20(3):561-582. https://doi.org/10.1142/S0218127410025892

Liu Y, Li BW, Lu JQ, et al., 2017. Pinning control for the disturbance decoupling problem of Boolean networks. *IEEE Trans Autom Contr*, 62(12):6595-6601. https://doi.org/10.1109/TAC.2017.2715181

Lu JQ, Zhong J, Ho DWC, et al., 2016. On controllability of delayed Boolean control networks. *SIAM J Contr Optim*, 54(2):475-494. https://doi.org/10.1137/140991820

Lu JQ, Sun LJ, Liu Y, et al., 2018a. Stabilization of Boolean control networks under aperiodic sampled-data control. *SIAM J Contr Optim*, 56(6):4385-4404. https://doi.org/10.1137/18M1169308

Lu JQ, Li ML, Huang TW, et al., 2018b. The transformation between the Galois NLFSRs and the Fibonacci NLFSRs via semi-tensor product of matrices. *Automatica*, 96:393-397. https://doi.org/10.1016/j.automatica.2018.07.011

Meng M, Lam J, Feng JE, et al., 2019. Stability and stabilization of Boolean networks with stochastic delays. *IEEE Trans Autom Contr*, 64(2):790-796. https://doi.org/10.1109/TAC.2018.2835366

Sun LJ, Lu JQ, Ching WK, 2020. Switching-based stabilization of aperiodic sampled-data Boolean control networks with all subsystems unstable. *Front Inform Technol Electron Eng*, 21(2):260-267. https://doi.org/10.1631/FITEE.1900312

Tong LY, Liu Y, Li YY, et al., 2018. Robust control invariance of probabilistic Boolean control networks via event-triggered control. *IEEE Access*, 6:37767-37774. https://doi.org/10.1109/ACCESS.2018.2828128

Veliz-Cuba A, Stigler B, 2011. Boolean models can explain bistability in the *lac* operon. *J Comput Biol*, 18(6):783-794. https://doi.org/10.1089/cmb.2011.0031

Wang B, Feng JE, Meng M, 2019. Model matching of switched asynchronous sequential machines via matrix approach. *Int J Contr*, 92(10):2430-2440. https://doi.org/10.1080/00207179.2018.1441552

Wu YH, Sun XM, Zhao XD, et al., 2019. Optimal control of Boolean control networks with average cost: a policy iteration approach. *Automatica*, 100:378-387. https://doi.org/10.1016/j.automatica.2018.11.036

Yu YY, Feng JE, Pan JF, 2019a. Block decoupling of Boolean control networks. *IEEE Trans Autom Contr*, 64(8):3129-3140. https://doi.org/10.1109/TAC.2018.2880411

Yu YY, Wang B, Feng JE, 2019b. Input observability of Boolean control networks. *Neurocomputing*, 333:22-28. https://doi.org/10.1016/j.neucom.2018.12.014

Zhong J, Ho DWC, Lu JQ, et al., 2019. Pinning controllers for activation output tracking of Boolean network under one-bit perturbation. *IEEE Trans Cybern*, 49(9):3398-3408. https://doi.org/10.1109/TCYB.2018.2842819

Zhu QX, Liu Y, Lu J, et al., 2019. Further results on the controllability of Boolean control networks. *IEEE Trans Autom Contr*, 64(1):440-442. https://doi.org/10.1109/TAC.2018.2830642

Zhu SY, Lou JG, Liu Y, et al., 2018. Event-triggered control for the stabilization of probabilistic Boolean control networks. *Complexity*, 2018:9259348. https://doi.org/10.1155/2018/9259348

Zhu SY, Lu JQ, Liu Y, 2019a. Asymptotical stability of probabilistic Boolean networks with state delays. *IEEE Trans Autom Contr*, in press. https://doi.org/10.1109/TAC.2019.2934532

Zhu SY, Lu JQ, Liu Y, et al., 2019b. Output tracking of probabilistic Boolean networks by output feedback control. *Inform Sci*, 483:96-105. https://doi.org/10.1016/j.ins.2018.12.087