SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

January 2020, Vol. 63 112203:1–112203:11
https://doi.org/10.1007/s11432-019-9898-3

Stabilization of logical control networks: an event-triggered control approach

Shiyong ZHU[1,2], Yang LIU[2*], Yijun LOU[3] & Jinde CAO[1*]

[1]*Jiangsu Provincial Key Laboratory of Networked Collective Intelligence, School of Mathematics, Southeast University, Nanjing* 210096*, China;*
[2]*College of Mathematics and Computer Science, Zhejiang Normal University, Jinhua* 321004*, China;*
[3]*Department of Applied Mathematics, Hong Kong Polytechnic University, Hong Kong, China*

Received 24 January 2019/Revised 17 March 2019/Accepted 29 April 2019/Published online 24 December 2019

Abstract This paper investigates the global stabilization problem of k-valued logical control networks (KVLCNs) via event-triggered control (ETC), where the control inputs only work at several certain individual states. Compared with traditional state feedback control, the designed ETC approach not only shortens the transient period of logical networks but also decreases the number of controller executions. The content of this paper is divided into two parts. In the first part, a necessary and sufficient criterion is derived for the event-triggered stabilization of KVLCNs, and a construction procedure is developed to design all time-optimal event-triggered stabilizers. In the second part, the switching-cost-optimal event-triggered stabilizer is designed to minimize the number of controller executions. A labeled digraph is obtained based on the dynamic of the overall system. Utilizing this digraph, we formulate a universal and unified procedure called the minimal spanning in-tree algorithm to minimize the triggering event set. Furthermore, we illustrate the effectiveness of obtained results through several numerical examples.

Keywords logical control network, event-triggered control, stabilization, semi-tensor product, minimal spanning in-tree

Citation Zhu S Y, Liu Y, Lou Y J, et al. Stabilization of logical control networks: an event-triggered control approach. Sci China Inf Sci, 2020, 63(1): 112203, https://doi.org/10.1007/s11432-019-9898-3

1 Introduction

Recently, the rapid development of DNA microarrays has set the stage for mathematical modeling of genetic regulatory networks [1]. Generally, numerical formal types of mathematical models have been proposed to depict, simulate, and even predict the dynamic behavior of biological networks, such as Markov-type genetic networks [2] and Boolean networks (BNs). Based on experimental results, BN models, which were originally proposed by Kauffman in 1969 [3], have been capable of forecasting the dynamic sequence of protein-activation patterns within genetic regulation networks [4]. A typical biological application is the cell cycle control network in yeast [5]. In addition, the modality of BNs has constructed a natural framework to ensure detailed comprehension and provide insights into the dynamic behavior exhibited by large-scale genetic regulation networks.

In a Boolean model, the expression of each node on a network is approximated by two levels, namely, 1 (ON) and 0 (OFF). The state update of each gene is determined by a pre-assigned logical function associated with the states of in-neighbor genes. As mentioned in [6], a recent significant discovery in systems biology is that exogenous perturbations, which can be described as "control", are almost

* Corresponding author (email: liuyang@zjnu.edu.cn, jdcao@seu.edu.cn)

ubiquitous in many biological systems. Thus, the concept of Boolean control networks (BCNs) has been formally generated by adding binary inputs to the BNs [7]. In capillary endothelial cells [8], for instance, a simple BCN has been established to simulate the dynamic behavior of the signaling system, where growth factors and cell shape (spreading) are both presented by two external inputs.

Since these extensive applications for therapeutic interventions in the field of systems biology, the investigations on BNs and BCNs have attracted widespread attention among scientists and scholars. Although BNs and BCNs seem simple, a systematic and valid tool to analyze and control these Boolean models was still lacking. Recently, Cheng et al. [9] have presented a generalized matrix product called semi-tensor product (STP). Owing to this technique, a considerable amount of studies have been inspired and several classical control problems have been extensively investigated in BCNs, including but not limited to stabilization and set stabilization [10–20], reachability and controllability [21–23], observability [24], synchronization [25, 26], function perturbation [27], as well as several other problems [28–32]. The main idea of the STP approach is to convert the logical dynamic of a BN (BCN) into a normal discrete-time linear (bilinear) system. Furthermore, the knowledge of matrix theory and graph theory can be applied to systematically analyze BNs and BCNs. Additionally, STP of matrices also provides convenience for analyzing k-valued logical control networks (KVLCNs), non-linear shift registers, finite automata and so on [33]. Indeed, KVLCNs, which can be regarded as a generalization of traditional BCNs to a certain extent, are more complex and have wider applications than BCNs. For example, the number of feasible choices in each player's action set may be more than two for a finite evolutionary networked game as described in [34, 35], but a binary Boolean model cannot describe such a case. Therefore, considering KVLCNs is an important task for the present study.

The controller design strategy is always an interesting topic in complex networks [36–38], naturally in genetic regulatory networks. Numerical control schemes have been developed in the study of logical systems, including but not limited to state feedback control [11], output feedback control [10], pinning control [21, 39], and sampled-data control [40]. Unfortunately, in the aforementioned control paradigms, the control inputs need to be executed at each time instant; it is a waste of resources if the dynamic evolution of the original network is desirable. Motivated by this problem, another alternative control paradigm called event-triggered control (ETC) has been proposed in [41]. With the advent of this triggering mechanism, the control cost can be reduced substantially; thus, ETC has been applied extensively in the study of logical control systems [34, 42–44], multi-agent systems [45], and smart grids [46]. Following the main stream of research, this study considers the event-triggered controller, which was an intermittent control strategy first proposed in the late 1990s [41]. As reported in [34, 42, 44], this type of ETC consists of two parts: (1) a state feedback mechanism to determine the control inputs and (2) a set of states to decide when the control inputs should be considered.

To date, ETC has been formally used to address many problems of KVLCNs. In [42], two classes of event-triggered controllers have been designed to deal with the disturbance decoupling problem of BCNs, and several necessary and sufficient conditions have been derived for determining whether this problem can be solved. Moreover, an effective event-triggered approach has been developed to achieve the global stabilization of finite evolutionary networked games by some reachable sets with respect to the designated state [34]. Meanwhile, the number of control executions has been minimized by an adjustment algorithm in some special circumstances. However, because the structure of the alternative control system has been given beforehand, the approach reported in [34] may become invalid, as shown in Example 2 introduced in this paper. Inspired by this situation, we believe that a universal and unified approach should be developed to minimize the number of controller executions. Thereafter, this type of event-triggered controller has been further generalized to investigate the global stabilization of probabilistic BCNs [44]. Moreover, the design of the time-optimal event-triggered stabilizer remains open.

In this study, the global stabilization problem of KVLCNs is addressed via the time-optimal event-triggered controller and switching-cost-optimal event-triggered controller. The main contributions of this study are listed as follows:

• In the first part of the study, the time-optimal event-triggered stabilizer is designed. Through STP technique, the algebraic framework of the KVLCN under ETC is established; it consists of a network inherent transition matrix, an alternative network transition matrix, and a triggering event set. Similar to the time-optimal state feedback stabilizer in [11], a necessary and sufficient criterion is derived for the event-triggered stabilization. Furthermore, a constructive procedure is developed to design all time-optimal event-triggered stabilizers.

- In the second part, the switching-cost-optimal stabilizer, which is event-triggered and has a minimal number of controller executions, is designed. The labeled digraph is constructed to describe the dynamic behavior of the event-triggered controlled KVLCN. Moreover, based on knowledge of graph theory, the number of controller executions is minimized through a universal procedure called minimal spanning in-tree algorithm. It deserves formulating that this algorithm can handle all circumstances and overcome the constraint of the method in [34].

The rest of this paper is structured as follows. Some preliminaries are introduced in Section 2. Section 3 presents the main results of the study, and proposes several illustrative examples to demonstrate the effectiveness of the results. A brief conclusion is provided in Section 4.

2 Preliminaries and problem formulation

In this section, some necessary preliminaries are presented.

- $\mathbb{N}$ and $\mathbb{R}$ are the sets of all natural integers and real integers, respectively.
- $\mathbb{R}_{m\times n}$ is the set of all $m \times n$ real matrices.
- $[a, b]$ represents the set of all integers between a and b, where $a, b \in \mathbb{N}$.
- $\mathscr{D}_k := [0, k-1]$.
- $\mathscr{D}_k^n := \underbrace{\mathscr{D}_k \times \mathscr{D}_k \times \cdots \times \mathscr{D}_k}_{n}$.
- $\mathrm{Col}_i(A)$ ($\mathrm{Row}_i(A)$) is the ith column (row) of matrix A.
- $\delta_n^i := \mathrm{Col}_i(I_n)$, where I_n is the $n \times n$ identity matrix.
- Δ_n is the set of all columns of identity matrix I_n.
- $A \in \mathbb{R}_{m\times n}$ is called a Boolean matrix, if $[A]_{ij} \in \mathscr{D}_2$, where the (i, j)th element of matrix A is denoted by $[A]_{ij}$.
- $A \in \mathbb{R}_{m\times n}$ is called a logical matrix, if $A = [\delta_m^{i_1}, \delta_m^{i_2}, \dots, \delta_m^{i_n}]$, simply represented as $\delta_m[i_1, i_2, \dots, i_n]$.
- $\mathscr{L}_{m\times n}$ consists of all $m \times n$ logical matrices.
- $|S|$ is the cardinal number of set S.

2.1 STP of matrices

Definition 1 (Cheng et al. [9]). The STP of $A \in \mathbb{R}_{m\times n}$ 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 α is the least common multiple of n and p, and '$\otimes$' is the Kronecker product of matrices.

Remark 1. As a generalization of conventional matrix product, when $n = p$, $A \ltimes B = (A \otimes I_1)(B \otimes I_1) = AB$. The STP of matrices provides a method to multiply two matrices with arbitrary dimensions (see [9] for further details). In general, symbol "$\ltimes$" is omitted without any confusion.

Lemma 1 (Cheng et al. [9]). Swap matrix $W_{[m,n]}$ is an $mn \times mn$ logical matrix defined as $W_{[m,n]} = \left[I_n \otimes \delta_m^1, \dots, I_n \otimes \delta_m^m\right]$. Based on $W_{[m,n]}$, the pseudo-commutative law of STP is concluded as follows:

(1) If $X \in \mathbb{R}_{m\times 1}$ and $B \in \mathbb{R}_{p\times q}$, then $XB = (I_m \otimes B)X$.

(2) If $X \in \mathbb{R}_{m\times 1}$ and $Y \in \mathbb{R}_{n\times 1}$, then $YX = W_{[m,n]}XY$.

Supposing that $\boldsymbol{x} = (x_1, x_2, \dots, x_n) \in \mathscr{D}_k^n$, we define a bijection $\Gamma_n : \mathscr{D}_k^n \to \Delta_{k^n}$ as

$$\Gamma_n(\boldsymbol{x}) = \ltimes_{i=1}^n \delta_k^{k-x_i} = \delta_{k^n}^s,$$

where $s = \sum_{i=1}^n (k - x_i - 1)k^{n-i}$ and $\Gamma_n(\boldsymbol{x})$ is called the equivalent delta form of vector $\boldsymbol{x}$. Then, arbitrary logical function $f : \mathscr{D}_k^n \to \mathscr{D}_k$ can be expressed in its algebraic representation by the following lemma.

Lemma 2 (Cheng et al. [9])**.** For a logical function $f(x_1, x_2, \ldots, x_n) : \mathscr{D}_k^n \to \mathscr{D}_k$, a determined matrix $M_f \in \mathscr{L}_{k\times k^n}$ exists, which is called the structure matrix of f, such that

$$f(x_1, x_2, \ldots, x_n) = M_f \Gamma_n (x_1, x_2, \ldots, x_n).$$

2.2 Dynamics of KVLCNs under event-triggered controllers

The KVLCN under ETC, presented as follows, consists of an inherent non-control k-valued logical network (KVLN) (1a), an alternative KVLCN (1b), and a triggering event set $\Lambda \subseteq \mathscr{D}_k^n$ standing for certain individual states where the control inputs are triggered:

$$\begin{cases} x_1(t+1) = f_1(x_1(t), \ldots, x_n(t)), \\ x_2(t+1) = f_2(x_1(t), \ldots, x_n(t)), \\ \qquad \vdots \\ x_n(t+1) = f_n(x_1(t), \ldots, x_n(t)), \end{cases} \tag{1a}$$

$$\begin{cases} x_1(t+1) = f'_1(x_1(t), \ldots, x_n(t), u_1(t), \ldots, u_m(t)), \\ x_2(t+1) = f'_2(x_1(t), \ldots, x_n(t), u_1(t), \ldots, u_m(t)), \\ \qquad \vdots \\ x_n(t+1) = f'_n(x_1(t), \ldots, x_n(t), u_1(t), \ldots, u_m(t)), \end{cases} \tag{1b}$$

where $f_i : \mathscr{D}_k^n \to \mathscr{D}_k$ and $f'_i : \mathscr{D}_k^{n+m} \to \mathscr{D}_k$, $i \in [1, n]$, are logical functions, and $x_i \in \mathscr{D}_k$, $u_j \in \mathscr{D}_k$, $j \in [1, m]$, are states and control inputs, respectively.

The ETC mechanism is essentially an intermittent control strategy. In particular, when the dynamic of inherent system (1a) evolves desirably, the system is maintained in the form of (1a) and the control inputs are not triggered. Otherwise, it means that the system's state locates in the set Λ, KVLCN (1b) works, and the control inputs are considered.

Then, we present the equivalent algebraic expression of the event-triggered controlled KVLCN. To facilitate the analysis, let $x(t) = \Gamma_n (x_1(t), x_2(t), \ldots, x_n(t)) \in \Delta_N$ and $u(t)=\Gamma_m (u_1(t), u_2(t), \ldots, u_m(t)) \in \Delta_M$, where $N = k^n$ and $M = k^m$. KVLN (1a) and KVLCN (1b) can be algebraically represented by Lemma 2 as follows:

$$x_i(t+1) = M_i x(t), \; i \in [1, n], \tag{2a}$$

$$x_j(t+1) = M'_j u(t) x(t), \; j \in [1, n], \tag{2b}$$

where $M_i \in \mathscr{L}_{k\times N}$ and $M'_j \in \mathscr{L}_{k\times MN}$. Then, the equations in (2a) and (2b) are further multiplied to show that

$$x(t+1) = Lx(t), \tag{3a}$$

$$x(t+1) = L'u(t)x(t), \tag{3b}$$

where $L=M_1 * M_2 * \cdots * M_n \in \mathscr{L}_{N\times N}$ is the inherent transition matrix, and $L' = M'_1 * M'_2 * \cdots * M'_n \in \mathscr{L}_{N\times MN}$ is the transition matrix of alternative subsystem (1b), where '$*$' is the Khatri-Rao product [47].

Therefore, if we define $\Gamma(\Lambda) := \{\Gamma_n(\boldsymbol{x}) : \boldsymbol{x} \in \Lambda\}$, the overall dynamic of the KVLCN with ETC can be synoptically described as

$$x(t+1) = \begin{cases} Lx(t), & x(t) \in \Delta_N \backslash \Gamma(\Lambda), \\ L'u(t)x(t), & x(t) \in \Gamma(\Lambda). \end{cases} \tag{4}$$

Or equivalently, the above dynamic can be given as

$$x(t+1) = [L, L']\, \widetilde{u}(t) x(t) := \widetilde{L} \widetilde{u}(t) x(t), \tag{5}$$

where the novel control $\widetilde{u}(t) \in \Delta_{M+1}$ is constructed from $u(t)$ as follows: (1) If $x(t) \in \Delta_N \backslash \Gamma(\Lambda)$, then $\widetilde{u}(t) := \delta_{M+1}^1$. (2) If $x(t) \in \Gamma(\Lambda)$, then one obtains that $\widetilde{u}(t) := [0, u(t)^{\mathrm{T}}]^{\mathrm{T}}$. Here and elsewhere, 'T' is the transpose of the matrix.

The state trajectory of system (5) with $x(0; x_0, \widetilde{\boldsymbol{u}}) = x_0$ with respect to a certain control sequence $\widetilde{\boldsymbol{u}} : \{0, 1, 2, \ldots\} \rightarrow \Delta_{M+1}$ is recorded as $x(t; x_0, \widetilde{\boldsymbol{u}})$. Then, the concept of the global event-triggered stabilization for system (5) with respect to $x^* \in \Delta_N$ is presented, where x^* is supposed to be δ_N^r without loss of generality.

Definition 2. For a given state $\delta_N^r \in \Delta_N$, system (5) is said to be globally stabilizable to δ_N^r, i.e., δ_N^r-stabilization, if for every $x_0 \in \Delta_N$, a positive integer T and a control sequence $\widetilde{\boldsymbol{u}} : \{0, 1, 2, \ldots\} \rightarrow \Delta_{M+1}$ exist such that $t \geqslant T$ implies $x(t; x_0, \widetilde{\boldsymbol{u}})=\delta_N^r$.

Remark 2. Because system (5) contains the information of the triggering event set $\Gamma(\Lambda)$, the stabilization of system (5) also can be called the event-triggered stabilization of system (4). Without raising any confusion, we simply refer to "stabilization" in the following sections.

In this study, the control input $u(t)$ in (4) is considered as the feedback of state $x(t)$, that is,

$$u(t) = Gx(t) = \delta_M[\beta_1, \beta_2, \ldots, \beta_N]x(t), \tag{6}$$

where $G \in \mathscr{L}_{M\times N}$ is called the state feedback matrix. In response to (6), $\widetilde{u}(t)$ also can be regarded as a special feedback of $x(t)$ with the "state feedback matrix" $\widetilde{G}$, namely, $\widetilde{u}(t) = \widetilde{G}x(t)$. In detail, $\widetilde{G} = \delta_{M+1}[\gamma_1, \gamma_2, \ldots, \gamma_N]$ is built as

$$\gamma_j = \begin{cases} 1, & \delta_N^j \in \Delta_N \backslash \Gamma(\Lambda), \\ \beta_j + 1, & \delta_N^j \in \Gamma(\Lambda). \end{cases} \tag{7}$$

The objective of this paper is to design the possible state feedback matrix $\widetilde{G} \in \mathscr{L}_{(M+1)\times N}$ such that KVLCN (5) is globally stabilizable to δ_N^r under two classes of event-triggered controllers, that is, the time-optimal stabilizer and the switching-cost-optimal stabilizer. Here, the time-optimal stabilizer aims to minimize the transient period and the switching-cost-optimal stabilizer aims to minimize the cardinal number of triggering event set $|\Gamma(\Lambda)|$.

3 Main results

In this section, we first develop the event-triggered controller for the minimum-time stabilization of KVLCN (5), which can also be called the time-optimal stabilizer. In the second part of this study, we develop an event-triggered controller with the minimal triggering event set, called the switching-cost-optimal stabilizer, under the framework of labeled digraph.

3.1 Design of time-optimal event-triggered stabilizer

In this subsection, the time-optimal event-triggered stabilizers are designed. We consider a v-step reachable set with respect to state δ_N^r that is defined as in [11]:

$$\mathscr{R}_v(r) = \Big\{\delta_N^j \in \Delta_N : \text{A control sequence } \widetilde{\boldsymbol{u}}(0), \widetilde{\boldsymbol{u}}(1), \ldots, \widetilde{\boldsymbol{u}}(v-1) \in \Delta_{M+1} \\ \text{exists such that } x(v; \delta_N^j, \widetilde{\boldsymbol{u}}(0), \widetilde{\boldsymbol{u}}(1), \ldots, \widetilde{\boldsymbol{u}}(v-1)) = \delta_N^r\Big\}. \tag{8}$$

On the basis of $\mathscr{R}_v(r)$ defined above, the following theorem can be obtained; its proof is straightforward and omitted.

Theorem 1. For a given state $\delta_N^r \in \Delta_N$, system (4) can be globally δ_N^r-stabilized by an event-triggered controller if and only if both of the following conditions are satisfied:

(1) $\delta_N^r \in \mathscr{R}_1(r)$;

(2) An integer $l \in [1, N-1]$ exists such that $\mathscr{R}_l(r) = \Delta_N$.

Without any confusion, the minimal integer satisfying condition (2) is denoted by l^*. Based on the assumption that conditions (1) and (2) in Theorem 1 are satisfied, the implication is that $\mathscr{R}_{i+1}(r) \supseteq \mathscr{R}_i(r)$

for all $i \in [0, l^*-1]$, where $\mathscr{R}_0(r) = \{\delta_N^r\}$. Then, we aim to develop a constructive procedure for the "state feedback matrix" $\widetilde{G}$, under which the transient period of (5) is minimal.

To this end, we split Δ_N into mutually disjoint sets as

$$\Delta_N = (\mathscr{R}_{l^*}(r) \setminus \mathscr{R}_{l^*-1}(r)) \cup \cdots \cup (\mathscr{R}_2(r) \setminus \mathscr{R}_1(r)) \cup (\mathscr{R}_1(r) \setminus \mathscr{R}_0(r)) \cup \mathscr{R}_0(r). \tag{9}$$

To each $\delta_N^i \in \Delta_N$, a unique integer $l_i \in [1, l^*]$ can be found such that $\delta_N^i \in \mathscr{R}_{l_i}(r) \setminus \mathscr{R}_{l_i-1}(r)$. Define $\alpha_i = \widetilde{L}\delta_{MN}^i$ for $i \in [1, MN]$. The 'state feedback matrix' $\widetilde{G} = \delta_{M+1}[\gamma_1, \gamma_2, \ldots, \gamma_N]$ can be given by the following procedure:

(1) If $\alpha_r = r$, let $\gamma_r = 1$. Otherwise, namely, $\alpha_r \neq r$, let γ_r be a solution of $\alpha_{(\gamma_r-1)N+r} = r$.

(2) For $i \in [1, N]\backslash\{r\}$, if $\delta_N^{\alpha_i} \in \mathscr{R}_{l_i-1}(r)$, let $\gamma_i = 1$. Otherwise, let γ_i be a solution of $\delta_N^{\alpha_{(\gamma_i-1)N+i}} \in \mathscr{R}_{l_i-1}(r)$.

Remark 3. Under the constructed controller, all states in Δ_N can reach δ_N^r after at most l^* steps. This time-optimal event-triggered stabilizer simultaneously reduces the control inputs as much as possible. If all time-optimal stabilizers are necessary, we only need to modify the above procedure trivially. Therefore, we ignore it here.

Once matrix $\widetilde{G}$ is obtained, the triggering event set $\Gamma(\Lambda)$ can immediately be calculated as $\{\delta_N^i : \gamma_i = 1\}$, and the initial state feedback matrix G is equal to $G = \delta_M[\beta_1, \beta_2, \ldots, \beta_N]$, where $\beta_i = \gamma_i - 1$ if $\gamma_i \neq 1$ and β_i can be arbitrarily selected in $[1, M]$ for $\gamma_i = 1$.

Example 1. Let $L = \delta_4[1, 1, 3, 4]$ and $L' = \delta_4[1, 3, 4, 1, 1, 3, 4, 1]$. We construct a novel system in the form of (5) with state transition matrix:

$$\widetilde{L} = \delta_4[1, 1, 3, 4, 1, 3, 4, 1, 1, 3, 4, 1]. \tag{10}$$

Let $r = 1$. We can easily calculate that $\mathscr{R}_1(1) = \{\delta_4^1, \delta_4^2, \delta_4^4\}$ and $\mathscr{R}_2(1) = \Delta_4$. As $\delta_4^1 \in \mathscr{R}_1(1)$ and $\mathscr{R}_2(1) = \Delta_4$, this system can be globally stabilizable to δ_4^1 under ETC.

According to $\widetilde{L}$, let γ_i be as in the aforementioned procedure. One has $\gamma_1 = 1$, $\gamma_2 = 1$, $\gamma_3 = 2, 3$, and $\gamma_4 = 2, 3$. Correspondingly, we can take the triggering event set $\Gamma(\Lambda) = \{\delta_4^1, \delta_4^2\}$. β_1, β_2, β_3, and β_4 can be arbitrarily selected from $\{1, 2\}$.

As observed from the preceding calculation, the transient period is 2. If we use the traditional state feedback control as in [11], it will be 3. Thus, we can formulate that the designed event-triggered controllers reduce the control cost and transient period more efficiently than the traditional state feedback controllers do.

3.2 Design of switching-cost-optimal event-triggered stabilizer

In this subsection, we assume that the conditions in Theorem 1 are satisfied, and we focus on designing the event-triggered stabilizer with optimal switching cost. That is, to minimize the triggering event set $\Gamma(\Lambda)$. In [34], an adjustment method has been formulated to minimize the triggering event set $\Gamma(\Lambda)$ in some special cases. However, this method is not capable of addressing certain generalized senses such as Example 2. Thus, a universal and unified approach to design the switching-cost-optimal stabilizer is still valuable and meaningful.

Example 2. We consider a logical system (5) with transition matrices $L = \delta_8[4, 2, 1, 3, 6, 5, 8, 5]$ and $L' = \delta_8[4, 2, 1, 2, 6, 8, 3, 3, 4, 2, 1, 2, 6, 5, 3, 3]$. We can easily confirm that this network is globally δ_8^2-stabilized under ETC by Theorem 1.

According to the approach proposed in [34], we first draw the attractors and basis[1)] of KVLN with respect to L as in Figure 1.

1) Please refer to [9] for more details on attractors and the basis of KVLNs.

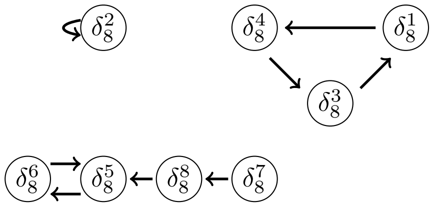

Figure 1 State transition graph of KVLN with respect to transition matrix $L = \delta_8[4, 2, 1, 3, 6, 5, 8, 5]$.

We can select a possible state in the attractors $\{\delta_8^1, \delta_8^3, \delta_8^4\}$ and $\{\delta_8^5, \delta_8^6\}$. Then, we add the feasible control inputs at these two states such that the overall system can be stabilized to δ_8^2. That is, the minimal number of controller execution is equal to 2.

However, as the transition matrix L' of KVLCN (3b) is determined, if δ_8^5 is selected, then it evolves to δ_8^6 under every control input. Otherwise, if δ_8^6 is selected, it evolves to δ_8^8 under $u = \delta_2^1$ or δ_2^2. Obviously, any state in the attractor $\{\delta_8^5, \delta_8^6\}$ cannot reach δ_8^2 by this approach. Thus, this method to minimize the triggering event set $\Gamma(\Lambda)$ is not applicable to this example.

Remark 4. In fact, as the transition matrix L' is known, considering L unilaterally is infeasible when minimizing $\Gamma(\Lambda)$ as in [34]. In the following, based on the knowledge of graph theory, we present a universal and unified approach to minimize the triggering event set.

First of all, a labeled digraph $\mathscr{G}$ is derived for equivalent graphical description of the dynamic of KVLCN (5). The labelled digraph $\mathscr{G}$ is an ordered pair (V, A) consisting of a set of vertices $V := [1, N]$ and a set of directed arcs A. For every arc $(i, j) \in A$, vertices i and j are respectively named as the starting and the ending vertices of arc (i, j).

For KVLN (3a), because $L \in \mathscr{L}_{N\times N}$ is a Boolean matrix, it can be associated with a labeled diagraph $\mathscr{G}_0 = (V, A_0)$. Here, A_0 is a real line arc set, where $\mathscr{G}_0$ has a real line arc (i, j) joining i to j if and only if $[L]_{ji} = 1$. For KVLCN (3b), L' is partitioned into $[L'_1, L'_2, \ldots, L'_M]$, where L'_μ, $\mu \in [1, M]$ are control-dependent transition matrices. Similar to the construction of $\mathscr{G}_0$, the labeled digraph for L_μ, denoted by $\mathscr{G}_\mu$, is associated with an order pair (V, A_μ), where A_μ is a dashed line arc set when a dashed line arc $(i, j) \in A_\mu$ if and only if $[L'_\mu]_{ji} = 1$. Furthermore, by uniting these labeled digraphs $\mathscr{G}_0$ and $\mathscr{G}_\mu$, $\mu \in [1, M]$, we can obtain the overall labelled digraph $\mathscr{G} = (V, A)$ as

$$\mathscr{G} = \bigcup_{\mu=0}^{M} \mathscr{G}_\mu = \left(V, \bigcup_{\mu=0}^{M} A_\mu\right).$$

Remark 5. In fact, the arc set A consists of some real line and dashed line arcs corresponding to the dynamics of (3a) and (3b), respectively. For convenience, they are denoted by an identical set A without distinction from notation. From the construction of $\mathscr{G}$, we can easily find that more than one arc may exist in the same direction with the same starting and ending vertices. Operating on the labeled digraph $\mathscr{G}$ may cause unnecessary issues and high time complexity. Therefore, we construct pretreatment for $\mathscr{G}$ before presenting the algorithm.

To facilitate the analysis, some pretreatment is operated on the labelled digraph $\mathscr{G}$. The labelled digraph after pretreatment is also denoted by $\mathscr{G} := (V, A)$ for convenience, where A represents the arc set of the labelled digraph after pretreatment. The pretreatment is listed as follows.

(1) Delete all self loops.

(2) For all ordered pairs $(i, j) \in [1, N] \times [1, N]$ and $i \neq j$, we retain the arc with minimal weight joining i to j and delete the others. If two such arcs exist, we select the arbitrary one.

(3) Assign each dashed line arc joining i to j by a control set $u_{(i,j)} := \{\mu : [L_\mu]_{ji} = 1,\ \mu \in [1, M]\}$.

As mentioned in [48], the stabilization problem of KVLCN can be equivalently described by the existence of spanning in-tree with the designated vertex r, which is called the root of tree. Thus, an approach to find the switching-cost-optimal event-triggered stabilizer is exactly to find a spanning in-tree at root r with the minimal number of dashed line arcs in labeled digraph $\mathscr{G}$.

To this end, weights N and 1 are respectively assigned to each dashed line arc and real line. Let $w(u,v)$ denote the weight on every arc (i,j), and then let $\mathscr{G} := (V, A, W)$ denote the labeled digraph $\mathscr{G}$ with weight, where W is a set of weight $w(i,j)$ for all $(i,j) \in A$. The spanning in-tree at root r with the minimal sum of weight is called the minimal spanning in-tree of labeled digraph $\mathscr{G}$. In the graph theory, an effective algorithm has been proposed to find the minimal spanning in-tree; it is called Edmonds's algorithm [49]. Moreover, a universal and unified procedure is firstly derived for the switching-cost-optimal event-triggered stabilizer.

Remark 6. The time complexity of Algorithm 1 is $O(HN)$, where $N = k^n$ and $H = |A|$.

Algorithm 1 Minimal spanning in-tree algorithm

Step 1: Initialize $i := 0$, $V_0 := V$, $E_0 := A$ and $W_0 := W$. Designate vertex r as the root.

Step 2: Calculate $J_1 = \{(v, \theta(v)) : v \in V_0 \setminus \{r\}\}$, where an order pair $(v, \theta(v))$ is the minimal weight arc among all $(v,j) \in E_0$.

Step 3: Check whether directed cycles exist in (V_i, J_{i+1}). If so, then proceed to Step 4. Otherwise, proceed to Step 7.

Step 4: Contract every cycle $\mathscr{C}$ into one new vertex to obtain a new diagraph $(V_{i+1}, E_{i+1}, W_{i+1})$; the weight set W_{i+1} is updated from W_i as follows; then $i := i + 1$ and go to Step 5.

- If (u,v) is an arc joining cycle $\mathscr{C}$, its weight is kept unchanged.
- If (u,v) is an arc away cycle $\mathscr{C}$, its weight is reassigned as $w(u,v) - w(u,\theta(u))$.
- The weights of the other arcs are kept unchanged.

Step 5: Perform pretreatment for the novel labeled digraph (V_i, E_i, W_i).

Step 6: Calculate $J_{i+1} = \{(v, \theta(v)) : v \in V_i \setminus \{r\}\}$, where an order pair $(v, \theta(v))$ is the minimal weight arc among all $(v,j) \in E_i$. Then, return to Step 3.

Step 7: Expand the contracted cycles formed during the preceding phase in reverse order of their contraction and remove one arc from each cycle to form a spanning in-tree.

The returned minimal spanning in-tree in Algorithm 1 is denoted by $\mathscr{G}^0 = (V, A^0, W^0)$, where $A^0 \subseteq A$ and $W_0 \subseteq W$. Once $\mathscr{G}^0$ is obtained, the corresponding event-triggered controllers can be constructed immediately. For every arc set $D \subseteq A$, $\lfloor D \rfloor$ consists of the starting vertex of each arc in D.

Example 3. In the following, Example 2 is reconsidered by the approach presented in this subsection.

First, weights 8 and 1 are respectively assigned to the dashed-line and real-line arcs. The labeled digraph after pretreatment is presented as Figure 2.

Then, J_1 is calculated by Step 2 of Algorithm 1 as in Figure 3.

As (V_0, J_1) has cycles $\mathscr{C}_1=\{1,3,4\}$ and $\mathscr{C}_2=\{5,6\}$, we proceed to Step 4 of Algorithm 1. As drawn in Figure 4, cycles $\mathscr{C}_1$ and $\mathscr{C}_2$ are contracted into novel vertices U and V, respectively.

Consequently, Figure 5 is obtained by repeating Steps 5 and 6 of Algorithm 1. A cycle in (V_1, J_2) still exists. Thus, vertices V and 8 are further recontracted and the weights of arcs are updated by Step 4 in Figure 6. By repeating Steps 5 and 6, (V_2, J_3) is obtained without any cycle as shown in Figure 7.

Finally, using Step 7 of Algorithm 1, we expand the contracted cycles formed during the preceeding phase in reverse order of their contraction and remove one arc from each cycle to form a spanning in-tree. Therefore, arcs $(3,4)$, $(5,6)$ and $(5,8)$ are removed. The obtained minimal spanning in-tree $\mathscr{G}^0$ is presented as in Figure 8.

Based on the minimal spanning in-tree $\mathscr{G}^0$, using Algorithm 2, the corresponding triggering event set is designed as $\Gamma(\Lambda) = \{\delta_8^4, \delta_8^6, \delta_8^8\}$, and the possible state feedback matrices are $G = \delta_8[*, *, *, *, *, 1, *, *]$, where $*$ is 1 or 2.

Algorithm 2 Corresponding event-triggered controller design from minimal spanning in-tree

Step 1: Construct the triggering event set $\Gamma(\Lambda)$. If $[L]_{rr} = 1$, then $\Gamma(\Lambda) = \{\delta_N^i : i \in \lfloor A^0 \backslash A_0 \rfloor\}$. Otherwise, $\Gamma(\Lambda) = \{\delta_N^i : i \in \lfloor A^0 \backslash A_0 \rfloor \cup \{r\}\}$.

Step 2: Determine the state feedback matrix G. Let β_r be randomly selected in Δ_M if $r \notin \Gamma(\Lambda)$; else, $\beta_r = u_{(r,r)}$. For every $j \in [1,N] \backslash \{r\}$, if $j \in \Gamma(\Lambda)$, a unique integer $t_j \in [1,N]$ satisfies $(j, t_j) \in A^0$. Let β_j be an arbitrary integer in $u_{(j,t_j)}$. Otherwise, let β_j be an arbitrary integer in $[1,M]$. The feasible state feedback matrix can be designed as $G = \delta_M[\beta_1, \beta_2, \ldots, \beta_N]$.

According to Figure 8, the number of control executions is equal to 3. If we utilize the traditional state

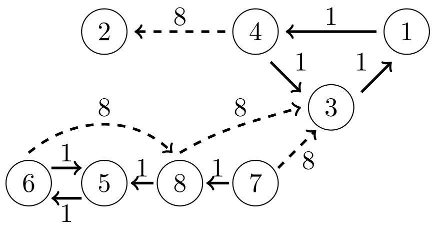

Figure 2 Labeled digraph after pretreatment (V_0, E_0, W_0).

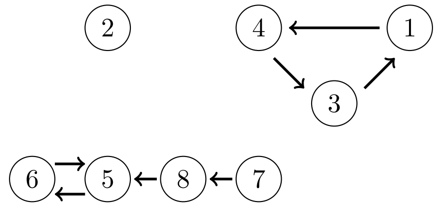

Figure 3 Calculate set J_1=$\{(v, \theta(v)) \mid v \in [1, 8]\}$ by Step 2 in Algorithm 1. That is, $\theta(1) = 4$, $\theta(3) = 1$, $\theta(4) = 3$, $\theta(5) = 6$, $\theta(6) = 5$, $\theta(7) = 8$ and $\theta(8) = 5$.

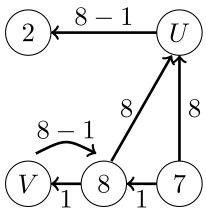

Figure 4 New constructed weighted directed graph (V_1, E_1, W_1). Based on Algorithm 1, $w(U, 2) = 8 - 1$ and $w(V, 8) = 8 - 1$. The weights of the other arcs remain unchanged.

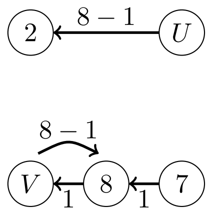

Figure 5 Find the set J_2 in Figure 4, where $J_2 = \{(V, 8), (8, V), (7, 8), (U, 2)\}$.

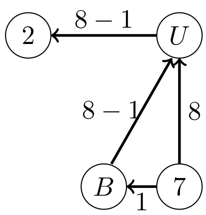

Figure 6 The vertices V and 8 are contracted into a novel vertex B. Let $w(U, B) = 8 - 1$ and the weights of the other arcs be unchanged.

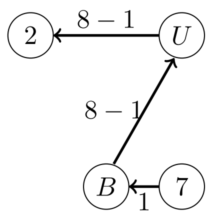

Figure 7 The set $J_3 = \{(2, U), (U, B), (B, 7)\}$.

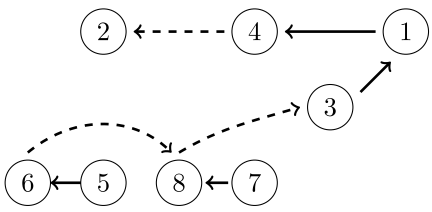

Figure 8 Minimal spanning in-tree $\mathscr{G}^0$ of Example 2.

feedback control, the number of control executions would be 7 in the transient period because all states need to be controlled.

4 Conclusion

This paper discussed the global stabilization problem of KVLCNs with ETC. By resorting to STP of matrices, we derived a necessary and sufficient condition for the global stabilization of event-triggered controlled KVLCNs. The corresponding time-optimal event-triggered stabilizer was achieved. In the latter part of the study, we designed the switching-cost-optimal event-triggered stabilizer. The labeled digraph of event-triggered controlled KVLCNs was constructed. By utilizing our knowledge of graph theory, we developed an effective algorithm called minimal spanning in-tree algorithm to minimize the number of control executions. This approach can address all circumstances and overcome the constraint faced by the existing method.

Acknowledgements This work was partially supported by National Natural Science Foundation of China (Grant Nos. 11671361, 61833005, 61573096), Natural Science Foundation of Zhejiang Province (Grant No. LD19A010001), Natural Science Foundation of Jiangsu Province (Grant No. BK20170019), and Jiangsu Provincial Key Laboratory of Networked Collective Intelligence (Grant No. BM2017002).

References

1 Davidson E H, Rast J P, Oliveri P, et al. A genomic regulatory network for development. Science, 2002, 295: 1669–1678
2 Liang J L, Lam J, Wang Z D. State estimation for Markov-type genetic regulatory networks with delays and uncertain mode transition rates. Phys Lett A, 2009, 373: 4328–4337
3 Kauffman S A. Metabolic stability and epigenesis in randomly constructed genetic nets. J Theor Biol, 1969, 22: 437–467
4 Goodwin B C. Temporal Organization in Cells: A Dynamic Theory of Cellular Control Processes. London: Academic, 1963
5 Davidich M I, Bornholdt S. Boolean network model predicts cell cycle sequence of fission yeast. PLoS One, 2008, 3: e1672
6 Ideker T, Galitski T, Hood L. A new approach to decoding life: systems biology. Annu Rev Genom Hum Genet, 2001, 2: 343–372
7 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
8 Huang S, Ingber D E. Shape-dependent control of cell growth, differentiation, and apoptosis: switching between attractors in cell regulatory networks. Exp Cell Res, 2000, 261: 91–103
9 Cheng D Z, Qi H S, Li Z Q. Analysis and Control of Boolean Networks: A Semi-Tensor Product Approach. Berlin: Springer, 2011
10 Bof N, Fornasini E, Valcher M E. Output feedback stabilization of Boolean control networks. Automatica, 2015, 57: 21–28
11 Li R, Yang M, Chu T G. State feedback stabilization for boolean control networks. IEEE Trans Automat Contr, 2013, 58: 1853–1857
12 Zhong J, Ho D W C, Lu J Q, et al. Global robust stability and stabilization of Boolean network with disturbances. Automatica, 2017, 84: 142–148
13 Wang L Q, Liu Y, Wu Z G, et al. Stabilization and finite-time stabilization of probabilistic Boolean control networks. IEEE Trans Syst Man Cybern Syst, 2019. doi: 10.1109/TSMC.2019.2898880
14 Li H T, Ding X Y. A control Lyapunov function approach to feedback stabilization of logical control networks. SIAM J Control Opt, 2019, 57: 810–831
15 Li Y Y, Li B W, Liu Y, et al. Set stability and stabilization of switched Boolean networks with state-based switching. IEEE Access, 2018, 6: 35624–35630
16 Meng M, Liu L, Feng G. Stability and l_1 gain analysis of Boolean networks with Markovian jump parameters. IEEE Trans Automat Contr, 2017, 62: 4222–4228
17 Li F F, Tang Y. Set stabilization for switched Boolean control networks. Automatica, 2017, 78: 223–230
18 Li B W, Lou J G, Liu Y, et al. Robust invariant set analysis of Boolean networks. Complexity, 2019, 2019: 1–8
19 Tong L Y, Liu Y, Li Y Y, et al. Robust control invariance of probabilistic boolean control networks via event-triggered control. IEEE Access, 2018, 6: 37767–37774
20 Gao Z G, Chen X D, Başar T. Stability structures of conjunctive Boolean networks. Automatica, 2018, 89: 8–20
21 Liu Y, Cao J D, Wang L Q, et al. On pinning reachability of probabilistic Boolean control networks. Sci China Inf Sci, 2020, 63: 169201
22 Zhu Q X, Liu Y, Lu J Q, et al. Further results on the controllability of Boolean control networks. IEEE Trans Automat Contr, 2019, 64: 440–442
23 Zhong J, Liu Y, Kou K I, et al. On the ensemble controllability of Boolean control networks using STP method. Appl Math Comput, 2019, 358: 51–62
24 Guo Y Q. Observability of Boolean control networks using parallel extension and set reachability. IEEE Trans Neural Netw Learn Syst, 2018, 29: 6402–6408
25 Li Y Y, Zhong J, Lu J Q, et al. On robust synchronization of drive-response Boolean control networks with disturbances. Math Probl Engin, 2018, 2018: 1737685
26 Chen H W, Liang J L. Local synchronization of interconnected Boolean networks with stochastic disturbances. IEEE Trans Neural Netw Learn Syst. 2019. doi: 10.1109/TNNLS.2019.2904978

27 Liu Y, Li B W, Chen H W, et al. Function perturbations on singular Boolean networks. Automatica, 2017, 84: 36–42
28 Wang S, Feng J E, Yu Y, et al. Further results on dynamic-algebraic Boolean control networks. Sci China Inf Sci, 2019, 62: 012208
29 Yu Y Y, Feng J E, Pan J F, et al. Block decoupling of Boolean control networks. IEEE Trans Automat Contr, 2019, 64: 3129–3140
30 Wu Y H, Sun X M, Zhao X D, et al. Optimal control of Boolean control networks with average cost: a policy iteration approach. Automatica, 2019, 100: 378–387
31 Chen S Q, Wu Y H, Macauley M, et al. Monostability and bistability of Boolean networks using semi-tensor products. IEEE Trans Control Netw Syst, 2018. doi:10.1109/TCNS.2018.2889015
32 Zhu Q X, Liu Y, Lu J Q, et al. On the optimal control of Boolean control networks. SIAM J Control Opt, 2018, 56: 1321–1341
33 Lu J Q, Li M L, Huang T W, et al. The transformation between the Galois NLFSRs and the Fibonacci NLFSRs via semi-tensor product of matrices. Automatica, 2018, 96: 393–397
34 Guo P L, Zhang H X, Alsaadi F E, et al. Semi-tensor product method to a class of event-triggered control for finite evolutionary networked games. IET Control Theor Appl, 2017, 11: 2140–2145
35 Mao Y, Wang L Q, Liu Y, et al. Stabilization of evolutionary networked games with length-r information. Appl Math Computat, 2018, 337: 442–451
36 Cao Y, Zhang L Y, Li C Y, et al. Observer-based consensus tracking of nonlinear agents in hybrid varying directed topology. IEEE Trans Cybern, 2017, 47: 2212–2222
37 Cao Y. Bifurcations in an Internet congestion control system with distributed delay. Appl Math Computat, 2019, 347: 54–63
38 Cao J D, Guerrini L, Cheng Z S. Stability and Hopf bifurcation of controlled complex networks model with two delays. Appl Math Comput, 2019, 343: 21–29
39 Liu Y, Li B W, Lu J Q, et al. Pinning control for the disturbance decoupling problem of Boolean networks. IEEE Trans Automat Contr, 2017, 62: 6595–6601
40 Lu J Q, Sun L J, Liu Y, et al. Stabilization of Boolean control networks under aperiodic sampled-data control. SIAM J Control Opt, 2018, 56: 4385–4404
41 Heemels W P M H, Johansson K H, Tabuada P. An introduction to event-triggered and self-triggered control. In: Proceedings of IEEE 51st Annual Conference on Decision and Control, Maui, 2012. 3270–3285
42 Li B W, Liu Y, Kou K I, et al. Event-triggered control for the disturbance decoupling problem of Boolean control networks. IEEE Trans Cybern, 2018, 48: 2764–2769
43 Li Y L, Li H T, Sun W W. Event-triggered control for robust set stabilization of logical control networks. Automatica, 2018, 95: 556–560
44 Zhu S Y, Lou J G, Liu Y, et al. Event-triggered control for the stabilization of probabilistic Boolean control networks. Complexity, 2018, 2018: 9259348
45 Tan X G, Cao J D, Li X D. Consensus of leader-following multiagent systems: a distributed event-triggered impulsive control strategy. IEEE Trans Cybern, 2019, 49: 792–801
46 Li C J, Yu X H, Yu W W, et al. Distributed event-triggered scheme for economic dispatch in smart grids. IEEE Trans Ind Inf, 2016, 12: 1775–1785
47 Ljung L, Söderström T. Theory and Practice of Recursive Identification. Cambridge: MIT Press, 1983
48 Liang J L, Chen H W, Liu Y. On algorithms for state feedback stabilization of Boolean control networks. Automatica, 2017, 84: 10–16
49 Edmonds J. Optimum branchings. J Res Natl Bureau Stan Sect B Math Math Phys, 1967, 71B: 233–240