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 feedback stabilizer design of Boolean networks based on network structure*

Jie ZHONG[1], Bo-wen LI[2,3], Yang LIU[‡1], Wei-hua GUI[4]

[1]*College of Mathematics and Computer Science, Zhejiang Normal University, Jinhua 321004, China*
[2]*School of Information Science and Engineering, Southeast University, Nanjing 210096, China*
[3]*School of Cyber Science and Engineering, Southeast University, Nanjing 210096, China*
[4]*School of Automation, Central South University, Changsha 410083, China*
E-mail: zhongjie0615@gmail.com; qfhxjy@126.com; liuyang@zjnu.edu.cn; gwh@csu.edu.cn

Received May 7, 2019; Revision accepted July 11, 2019; Crosschecked Sept. 12, 2019; Published online Oct. 14, 2019

Abstract: In genetic regulatory networks, a stable configuration can represent the evolutionary behavior of cell death or unregulated growth in genes. We present analytical investigations on output feedback stabilizer design of Boolean networks (BNs) to achieve global stabilization via the semi-tensor product method. Based on network structure information describing coupling connections among nodes, an output feedback stabilizer is designed to achieve global stabilization. Compared with the traditional pinning control design, the output feedback stabilizer design is not based on the state transition matrix of BNs, which can efficiently determine pinning control nodes and reduce computational complexity. Our proposed method is efficient in that the calculation of the state transition matrix with dimension $2^n \times 2^n$ is avoided; here n is the number of nodes in a BN. Finally, a signal transduction network and a D. melanogaster segmentation polarity gene network are presented to show the efficiency of the proposed method. Results are shown to be simple and concise, compared with traditional pinning control for BNs.

Key words: Boolean networks; Output feedback stabilizer; Network structure; Semi-tensor product of matrices
https://doi.org/10.1631/FITEE.1900229 **CLC number:** TP183

1 Introduction

As a typical formalism to model genetic regulatory networks, Boolean networks (BNs) have been widely studied over past decades (Kauffman, 1969; Kauffman et al., 2003). BNs are discrete-time logical systems, where nodes of networks can model genes. In a BN, each component is inter-connected and the logical evolution of each component is described by logical functions. Each node in a BN chooses a logical value from a logical set composed of variables 1 and 0. Thus, a BN associated with n nodes can be described by a sequence of functions $(f_1, f_2, \ldots, f_n) : \{1, 0\}^n \to \{1, 0\}^{n^2}$, where each function f_i is a logical function. The coupling connections among nodes of a BN are described by a digraph with vertices $\{x_1, x_2, \ldots, x_n\}$. An edge exists in the wiring digraph from vertex x_i to vertex x_j if function f_j is dependent on variable x_i, that is, if there exists a tuple $(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n) \in \{1, 0\}^{n-1}$ such that $f(x_1, \ldots, x_i, \ldots, x_n) \neq f(x_1, \ldots, \neg x_i, \ldots, x_n)$. Furthermore, one can define a sign for each edge: an edge $x_i \to x_j$ is positive if $f_j(x_1, x_2, \ldots, x_i = 0, x_{i+1}, \ldots, x_n) < f_j(x_1, x_2, \ldots, x_i = 1, x_{i+1}, \ldots, x_n)$ for some $(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n) \in \{1, 0\}^{n-1}$, and is negative if the inequality is reversed. If an elementary directed loop in the wiring digraph has an even (odd) number of negative edges, it is called

‡ Corresponding author
* Project supported by the National Natural Science Foundation of China (Nos. 61903339, 61321003, and 11671361) and the Zhejiang Provincial Natural Science Foundation of China (No. LD19A010001)
ORCID: Yang LIU, http://orcid.org/0000-0002-9005-9166

a positive (negative) loop.

In the past two decades, a number of researchers have sought to analyze and model BNs owing to the introduction of a new matrix product called the semi-tensor product (STP) of matrices (Cheng et al., 2010). In Fan et al. (2018), the structure of solutions to fuzzy relational equations was analyzed and all solutions to general decompositions of fuzzy relational equations can be computed. Because of its powerful applications, STP sets out a new direction for further studying and modeling BNs. In fact, the STP of matrices is a generalization of a common matrix product. It can be applied to a matrix product with arbitrary dimensions, and break the traditional dimension matching condition. One of the most important applications of STP is that it can be efficiently used as a powerful tool to analyze BNs, Petri networks, and other finite-valued systems. Thus, under the framework of bijective equivalence between logical functions and algebraic forms, much work has taken place (Laschov and Margaliot, 2012; Lu et al., 2018a, 2018b, 2018c; Wu and Shen, 2018; Zhu QX et al., 2018, 2019; Li BW et al., 2019c; Li XD et al., 2019). There is considerable literature on some classical topics of Boolean (control) networks, including observability (Fornasini and Valcher, 2013; Zhang and Zhang, 2016), synchronization (Li YY et al., 2017, 2018b), stability and stabilization (Li R et al., 2013; Kobayashi and Hiraishi, 2017; Li YY et al., 2018a; Mao et al., 2018; Meng et al., 2018; Li BW et al., 2019a, 2019b; Pan et al., 2019), normalization and identification (Liu et al., 2017), and disturbance decoupling (Yang et al., 2013). In Yu et al. (2019), on the basis of canonical solutions of logical matrix equations, a Boolean control network (BCN) can be equivalently decomposed into a series of relatively independent small-scale ones, which contributes to breaking the limitation on the sizes of networks. In Wang and Feng (2019), two types of detectability for probabilistic BNs, weak detectability and strong detectability, were introduced and investigated by an algebraic expression called the data form, which could help in the diagnosis of a class of diseases. For details on potential applications of STP, please refer to Cheng and Liu (2016).

One of the most significant issues in the analysis of Boolean models of molecular networks is the determination of steady states and stable configuration (Ay et al., 2009; Mori and Mochizuki, 2017). This is because steady states represent the stable dynamical configurations and activity levels of genes. In recent years, work has increased on the stability and stabilization of BNs via the STP method. For example, in Cheng et al. (2011), two kinds of control inputs (open-loop control and state feedback control) have been studied to achieve stabilization. In Zhu and Lin (2019), an optimal event-triggered feedback control was proposed to stabilize BNs under the STP method. In addition, in Li YY et al. (2018a), set stability and set stabilization for switched BNs were studied based on the STP method and state-based switching signals. Considering the impact of probability in BNs, event-triggered control for stabilization of probabilistic BCNs was studied in Zhu SY et al. (2018) using the information of the state transition matrix, while in Tong et al. (2018), robust control invariance of probabilistic BCNs was studied based on event-triggered control. Li R et al. (2013) first proposed an efficient method to design state feedback stabilizers, and later Li and Wang (2017) proposed a constructive procedure to design all possible feedback stabilizers using the families of reachable sets. The issue of stability and stabilization was extended to set stability/stabilization, which has been studied based on invariant subsets in Guo et al. (2015).

However, in these related references, the designed controllers were applied to all nodes or randomly to some selected nodes of networks. An interesting finding is the introduction of pinning control technique. It has been proved as a new technique and a hot spot to study BNs (Li, 2015). For example, stabilization under state feedback pinning controllers has been addressed in Li (2015). A detailed algorithm to design pinning controllers was first proposed based on the state transition matrix of BNs (Li, 2015). Unfortunately, since the design of pinning controllers is based on a matrix with dimension $2^n \times 2^n$, any algorithm has an exponentially computational complexity. This is one of the main drawbacks of using the state transition matrix, which seems to be hard to implement on large-dimensional BNs. Specifically, when the size of networks is larger than 20, the traditional pinning control method based on the state transition matrix will be useless.

Recently, much attention has been paid to reducing the computational complexity and finding more efficient ways to design controllers to achieve stabilization. For example, several kinds of control actions including edge deletion and node deletion have been proposed to identify specific control targets in biological networks, like the p53-mdm2 network and the T-LGL survival signaling network (Murrugarra et al., 2016). In addition, several attempts have been made to disclose the relationships between stability and the network structure (NS) of BNs including positive/negative feedback loops (Aracena, 2008; Paulevé and Richard, 2012; Campbell and Albert, 2014). Robert (1986) discovered that if there are no loops (including positive/negative feedback loops) in the NS digraph of BNs, then BNs achieve stability with a unique steady state and no cyclic attractors. Thus, it is crucial to disclose the complex inter-relationship between NS and the design of pinning control. In this study, an output feedback stabilizer is presented to achieve global stabilization of BNs based on NS of BNs. Compared with the literature (Li and Wang, 2013; Bof et al., 2015), the traditional design of output feedback stabilizers is applied to all nodes of networks, based on the state transition matrix of BNs. To the best of our knowledge, far too little attention has been paid to the design of the output feedback stabilizer for global stabilization based on the NS of BNs.

The main contributions of this study are listed below:

1. We first propose an output feedback stabilizer design for global stabilization based on an acyclic NS describing the nodes' connection of BNs. In references on analyzing and designing controls of BNs, the information of the state transition matrix of BNs is necessary. Fortunately, this is avoided using the proposed method via NS information.

2. Based on the NS of BNs, the computational complexity will be reduced from $O(2^n \times 2^n)$ to $O(2 \times 2^n)$, where n is the number of nodes in BNs, compared with the traditional method. Hence, the method of designing output feedback pinning control can be implemented for some large-dimensional BNs.

3. By deleting a certain number of edges such that the NS is acyclic, global stability will be guaranteed. Then an output feedback pinning control is designed. This is partially imposed on the nodes of networks. Without using the state transition matrix, the design of pinning control is much easier to implement than in traditional design.

Some basic notations are given as follows: $N = \{1, 2, \ldots, n\}$, $\mathbf{1}_n = (1, 1, \ldots, 1)$, $\mathcal{D} = \{0, 1\}$, $\mathbb{R}^{m\times n}$ is a set of $m \times n$ real matrices, N^+ is the set of positive integers, and $[a, b] = \{a, a + 1, \ldots, b\}$, where $a < b$ and $a, b \in N^+$. $\boldsymbol{\delta}_n^i$ is the i^{th} column of identity matrix $\boldsymbol{I}_n$, $\Delta_n = \{\boldsymbol{\delta}_n^1, \boldsymbol{\delta}_n^2, \ldots, \boldsymbol{\delta}_n^n\}$ denotes the columns' set of $\boldsymbol{I}_n$, $\mathrm{Col}_i(\boldsymbol{A})$ is the i^{th} column of matrix $\boldsymbol{A}$, $\mathrm{Col}(\boldsymbol{A})$ denotes the set of columns, matrix $\boldsymbol{L} = [\boldsymbol{\delta}_n^{i_1}, \boldsymbol{\delta}_n^{i_2}, \ldots, \boldsymbol{\delta}_n^{i_s}]$ is called a logical matrix, simplified by $\boldsymbol{L} = \boldsymbol{\delta}_n[i_1, i_2, \ldots, i_s]$, and $\boldsymbol{\mathcal{L}}_{m\times n}$ is the set of logical matrices with dimension $m \times n$. Swap matrix $\boldsymbol{W}_{[m,n]} = [\boldsymbol{I}_n \otimes \boldsymbol{\delta}_m^1, \boldsymbol{I}_n \otimes \boldsymbol{\delta}_m^2, \ldots, \boldsymbol{I}_n \otimes \boldsymbol{\delta}_m^m]$. Power reducing matrix $\boldsymbol{\Phi}_{2^n} = \boldsymbol{\delta}_{2^{2n}}[1, 2^n + 2, (3 - 1)2^n + 3, \ldots, (2^n - 2)2^n + 2^n - 1, 2^{2n}]$.

2 Problem formulation

Consider the following BN with n nodes and p outputs:

$$\begin{cases} x_i(t+1) = f_i\left(x_1(t), x_2(t), \ldots, x_n(t)\right), & i \in [1, n], \\ y_j(t) = g_j\left(x_1(t), x_2(t), \ldots, x_n(t)\right), & j \in [1, p], \end{cases} \tag{1}$$

where $x_i \in \mathcal{D}$ and $y_j \in \mathcal{D}$ are the states and outputs of system (1), respectively. $f_i : \mathcal{D}^n \to \mathcal{D}$ and $g_j : \mathcal{D}^n \to \mathcal{D}$ are logical functions. The output feedback stabilization problem in this study is to design an output feedback stabilizer in the form of

$$u_j(t+1) = h_j\left(y_1(t), y_2(t), \ldots, y_p(t)\right), \; j \in [1, m] \tag{2}$$

such that system (1) under controller (2), that is,

$$\begin{cases} x_i(t+1) = f_i\left(x_1(t), x_2(t), \ldots, x_n(t), u_i(t)\right), \\ \qquad\qquad\qquad\qquad\qquad i \in [1, m], \\ x_j(t+1) = f_j\left(x_1(t), x_2(t), \ldots, x_n(t)\right), \\ \qquad\qquad\qquad\qquad\qquad j \in [m+1, n], \\ y_j(t) = g_j\left(x_1(t), x_2(t), \ldots, x_n(t)\right), \\ \qquad\qquad\qquad\qquad\qquad j \in [1, p], \end{cases} \tag{3}$$

will be globally stabilized. This implies that system (3) will have a unique steady state as the attractor.

In the following, we separately convert system (1) and output feedback controller (2) into equivalent algebraic forms. To this end, we first recall the definition and some properties of the STP of matrices.

Definition 1 (STP (Cheng et al., 2010)) Given two matrices $\boldsymbol{A} \in \mathbb{R}^{n\times m}$ and $\boldsymbol{B} \in \mathbb{R}^{p\times q}$, the STP of

$\boldsymbol{A}$ and $\boldsymbol{B}$, denoted by $\boldsymbol{A} \ltimes \boldsymbol{B}$, is defined as $\boldsymbol{A} \ltimes \boldsymbol{B} = (\boldsymbol{A} \otimes \boldsymbol{I}_{l/m})(\boldsymbol{B} \otimes \boldsymbol{I}_{l/p})$, where l is the least common multiple of m and p and $\otimes$ is the Kronecker product of matrices.

To facilitate using the STP method in BNs, identify "1" and "0" with vectors, $\mathbf{1} \sim \boldsymbol{\delta}_2^1$ and $\mathbf{0} \sim \boldsymbol{\delta}_2^2$, respectively. Then the following proposition is given:

Proposition 1 (Matrix expression of logical functions (Cheng et al., 2010)) Let $f(\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_n) : \mathcal{D}^n \rightarrow \mathcal{D}$ be a logical function. Then there exists a unique matrix $\boldsymbol{F} \in \boldsymbol{\mathcal{L}}_{2\times 2^n}$ such that $f(\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_n) = \boldsymbol{F} \ltimes \boldsymbol{a}_1 \ltimes \boldsymbol{a}_2 \ltimes \cdots \ltimes \boldsymbol{a}_n$. Here, $\boldsymbol{F}$ is called the structure matrix of logical function f.

According to Proposition 1, some common structure matrices of basic logical operators are obtained, such as negation "$\neg$," conjunction "$\wedge$," disjunction "$\vee$," conditional "$\rightarrow$," and bi-conditional "$\leftrightarrow$." The corresponding structure matrices are denoted by $\boldsymbol{M}_{\mathrm{n}} = \boldsymbol{\delta}_2[2, 1]$, $\boldsymbol{M}_{\mathrm{c}} = \boldsymbol{\delta}_2[1, 2, 2, 2]$, $\boldsymbol{M}_{\mathrm{d}} = \boldsymbol{\delta}_2[1, 1, 1, 2]$, $\boldsymbol{M}_{\mathrm{i}} = \boldsymbol{\delta}_2[1, 2, 1, 1]$, and $\boldsymbol{M}_{\mathrm{e}} = \boldsymbol{\delta}_2[1, 2, 2, 1]$.

Using the vector form of the logical variable and setting $\boldsymbol{x}(t) = \ltimes_{i=1}^n \boldsymbol{x}_i(t)$, $\boldsymbol{u}(t) = \ltimes_{i=1}^m \boldsymbol{u}_i(t)$, and $\boldsymbol{y}(t) = \ltimes_{i=1}^p \boldsymbol{y}_i(t)$, one can convert systems (1) and (2) into the following algebraic systems:

$$\begin{cases} \boldsymbol{x}(t+1) = \boldsymbol{L}\boldsymbol{x}(t), \\ \boldsymbol{y}(t) = \boldsymbol{G}\boldsymbol{x}(t), \end{cases} \tag{4}$$

where $\boldsymbol{L} = \boldsymbol{F}_1 * \boldsymbol{F}_2 * \cdots * \boldsymbol{F}_n \in \boldsymbol{\mathcal{L}}_{2^n\times 2^n}$ is the state transition matrix of BNs, $\boldsymbol{F}_i \in \boldsymbol{\mathcal{L}}_{2\times 2^n}$ $(i \in [1, n])$ are structure matrices of f_i, $\boldsymbol{G} = \boldsymbol{G}_1 * \boldsymbol{G}_2 * \cdots * \boldsymbol{G}_p \in \boldsymbol{\mathcal{L}}_{2^p\times 2^n}$ is the output transition matrix of BNs, $\boldsymbol{G}_i$ are structure matrices of g_i $(i \in [1, n])$, and $\boldsymbol{H}_i \in \boldsymbol{\mathcal{L}}_{2\times 2^n}$ $(i \in [1, p])$ are structure matrices of functions h_i. In addition, symbol "$*$" is the Khatri-Rao product, defined as given matrices $\boldsymbol{A} \in \mathbb{R}^{s\times t}$ and $\boldsymbol{B} \in \mathbb{R}^{l\times t}$, $\boldsymbol{A} * \boldsymbol{B} = [\mathrm{Col}_1(\boldsymbol{A}) \ltimes \mathrm{Col}_1(\boldsymbol{B}), \mathrm{Col}_2(\boldsymbol{A}) \ltimes \mathrm{Col}_2(\boldsymbol{B}), \ldots, \mathrm{Col}_t(\boldsymbol{A}) \ltimes \mathrm{Col}_t(\boldsymbol{B})]$.

Remark 1 Over the past few years, the pinning control method has been studied based on the information of the state transition matrix $\boldsymbol{L}$ of system (4) (Li, 2015, 2016). However, in Li (2015, 2016), the design of the traditional pinning control is always based on the state transition matrix with dimension $2^n \times 2^n$. Thus, all the algorithms based on the state transition matrix have an exponential complexity, as one needs to calculate a matrix with size $2^n \times 2^n$. It is a major drawback of this approach in regard to the computational complexity and has already attracted much attention (Zhao et al., 2016). In this study, the shortcoming of traditional pinning control design is partially overcome by using the NS of BNs.

Thus, our main objective is to design an output feedback stabilizer to achieve global stabilization for system (1). However, the issue of stable configuration with respect to a target equilibrium state is not the main objective of this study, which needs further work. The main tool is the STP method, and the NS of BNs is used to design an output feedback stabilizer without using the state transition matrix.

3 Main results

BNs (Kauffman, 1969; Kauffman et al., 2003) are discrete-time logical systems, where nodes of networks represent genes that are inter-connected and the dynamical evolution of each node is determined by logical functions. However, in some logical functions, some variables may be non-functional. For example, $x_1 \vee x_2$ is dependent on variables x_1 and x_2, while $(x_1 \wedge x_2) \vee (x_1 \wedge \neg x_2)$ is not dependent on variable x_2, because $(x_1 \wedge x_2) \vee (x_1 \wedge \neg x_2) = x_1$.

Thus, we should define the dependency of logical functions to determine functional/non-functional variables.

Definition 2 A logical function $f(x_1, x_2, \ldots, x_n) : \mathcal{D}^n \rightarrow \mathcal{D}$ is said to be dependent on variable x_i if there exists a tuple $\bar{x} \in \mathcal{D}^{n-1}$ such that $f(\bar{x}, x_i) \neq f(\bar{x}, \neg x_i)$, where $\bar{x} = (x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n)$.

Using the STP method, one can obtain that there exists

$$(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n) = \boldsymbol{\delta}_{2^{n-1}}^j, \quad j \in \{1, 2, \ldots, 2^{n-1}\}$$

such that

$$\begin{aligned} &\boldsymbol{M}_f \ltimes x_1 \ltimes \cdots \ltimes x_{i-1} \ltimes \boldsymbol{\delta}_2^1 \ltimes \cdots \ltimes x_n \\ \neq &\boldsymbol{M}_f \ltimes x_1 \ltimes \cdots \ltimes x_{i-1} \ltimes \boldsymbol{\delta}_2^2 \ltimes \cdots \ltimes x_n. \end{aligned}$$

Assume that $\boldsymbol{M}_f$ is the structure matrix of function f. Hence, one can further obtain that

$$\begin{aligned} &\boldsymbol{M}_f \ltimes \boldsymbol{W}_{[2,2^{i-1}]} \ltimes \boldsymbol{\delta}_2^1 \ltimes x_1 \ltimes \cdots \ltimes x_{i-1} \ltimes \cdots \ltimes x_n \\ \neq &\boldsymbol{M}_f \ltimes \boldsymbol{W}_{[2,2^{i-1}]} \ltimes \boldsymbol{\delta}_2^2 \ltimes x_1 \ltimes \cdots \ltimes x_{i-1} \ltimes \cdots \ltimes x_n. \end{aligned}$$

According to Definition 2, one can determine functional and non-functional variables in logical functions. Thus, one has a connection rule among nodes;

here, we call such a connection the NS of BNs. To describe such an NS, we introduce a wiring digraph describing the NS of BNs.

Definition 3 Consider a BN associated with a sequence of logical functions $f = (f_1, f_2, \ldots, f_n)$. The wiring digraph of a BN is a digraph denoted by $G = (V, E)$ with n vertices; that is, $V = \{x_1, x_2, \ldots, x_n\}$, where $x_1, x_2, \ldots, x_n$ are nodes of the network. An edge $x_i \to x_j$ exists in the wiring digraph $G = (V, E)$ if and only if function f_j is dependent on x_i.

Steady states represent cell types like cell death or unregulated growth (Saadatpour et al., 2011). Then the definitions for steady states and global stability of system (1) are introduced as follows:

Definition 4 (Steady states) Considering a BN with a state $x(t) \in \mathcal{D}^n$, $x(t)$ is called a steady state if $x(t+1) = x(t)$.

Definition 5 (Global stability) System (1) is said to be globally stable if there exists only one unique steady state as the attractor without other cyclic attractors.

In this study, we focus only on the stable configuration of system (1) but not on the stable configuration with respect to a target equilibrium state. Actually, one significant relationship between NS of BNs and global stability has been disclosed in Robert (1986), as shown in the following theorem:

Theorem 1 (Criterion for global stability (Robert, 1986)) System (1) is globally stable if the wiring digraph $G = (V, E)$ is acyclic.

Now, an output feedback pinning control design will be presented to achieve global stabilization, based on the wiring digraph $G = (V, E)$. If the wiring digraph is acyclic, then the system will be globally stable. This means that the system has only one unique steady state. Then in the following sequel, an output feedback pinning control will be designed by deleting a certain number of edges in wiring digraph $G = (V, E)$ to lead to an acyclic NS.

Using the depth-first search algorithm, one can obtain the cycles and fixed points in the wiring digraph $G = (V, E)$. Then we will use the concept of feedback arc set (FAS) to delete edges, such that the wiring digraph $G = (V, E)$ becomes acyclic. The definition of an FAS is given as follows:

Definition 6 (FAS (Bang-Jensen and Gutin, 2008)) An FAS is defined as a subset of edges containing at least one edge of every cycle in a directed network, and it is called a minimum FAS if its cardinality is the minimum. Therefore, the removal of the FAS renders the network acyclic.

Then consider the wiring digraph $G = (V, E)$ which is not acyclic, one can always find an FAS. Here, we assume that $\{e_1, e_2, \ldots, e_p\}$ $(e_1, e_2 \ldots, e_p \in E)$ is a feasible FAS in the wiring digraph. In the wiring digraph, an edge may be shared by distinct nodes. Then for each edge $e_1, e_2, \ldots, e_p \in E$, one can obtain its corresponding ending vertex, denoted by symbols $\boldsymbol{P}_+(e_1), \boldsymbol{P}_+(e_2), \ldots, \boldsymbol{P}_+(e_p) \in \{x_1, x_2, \ldots, x_n\}$. For example, suppose an edge is given by $e_1 = x \to y$. Then we have $\boldsymbol{P}_+(e_1) = y$. We further assume that $\bigcup_{j=1}^{p} \boldsymbol{P}_+(e_j) = \{x_{\gamma_1}, x_{\gamma_2}, \ldots, x_{\gamma_\tau}\}$, $\Sigma_\tau \triangleq \{\gamma_1, \gamma_2, \ldots, \gamma_\tau\}$, $\gamma_1, \gamma_2, \ldots, \gamma_\tau \in [1, n]$.

Thus, under the above assumptions, we will design an output feedback pinning control on nodes x_j $(j \in \Sigma_\tau)$, to achieve global stabilization. Assume that

$$\begin{cases} \bigcup_{j=\varepsilon_1}^{l_1^{\varepsilon_1}} \boldsymbol{P}_+(e_j) = \{x_{\gamma_1}\}, \\ \bigcup_{j=\varepsilon_1}^{l_1^{\varepsilon_1}} \boldsymbol{P}_-(e_j) := \{x_{\nu_1^1}, x_{\nu_1^2}, \ldots, x_{\nu_1^{\varepsilon_1}}\}, \\ \quad\vdots \\ \bigcup_{j=\varepsilon_\tau}^{l_\tau^{\varepsilon_\tau}} \boldsymbol{P}_+(e_j) = \{x_{\gamma_\tau}\}, \\ \bigcup_{j=\varepsilon_\tau}^{l_\tau^{\varepsilon_\tau}} \boldsymbol{P}_-(e_j) := \{x_{\nu_\tau^1}, x_{\nu_\tau^2}, \ldots, x_{\nu_\tau^{\varepsilon_\tau}}\}, \end{cases}$$

where $\nu_1^1 < \nu_1^2 < \ldots < \nu_1^{\varepsilon_1}$, ..., $\nu_\tau^1 < \nu_\tau^2 < \ldots < \nu_\tau^{\varepsilon_\tau}$. Consider nodes x_i $(i \in \Sigma_\tau)$ and corresponding logical functions $x_i(t+1) = f_i(x_1(t), x_2(t), \ldots, x_n(t))$. Transform the logical functions $x_i(t+1) = f_i(x_1(t), x_2(t), \ldots, x_n(t))$ by deleting variables $x_{\nu_1^1}, x_{\nu_1^2}, \ldots, x_{\nu_1^{\varepsilon_1}}, \ldots, x_{\nu_\tau^1}, x_{\nu_\tau^2}, \ldots, x_{\nu_\tau^{\varepsilon_\tau}}$ and deleting connected logical operators as follows:

$$\begin{cases} \hat{f}_{\gamma_1}\left(\hat{X}_{\gamma_1}(t)\right) \Leftarrow f_{\gamma_1}(x_1(t), x_2(t), \ldots, x_n(t)), \\ \hat{f}_{\gamma_2}\left(\hat{X}_{\gamma_2}(t)\right) \Leftarrow f_{\gamma_2}(x_1(t), x_2(t), \ldots, x_n(t)), \\ \quad\vdots \\ \hat{f}_{\gamma_\tau}\left(\hat{X}_{\gamma_\tau}(t)\right) \Leftarrow f_{\gamma_\tau}(x_1(t), x_2(t), \ldots, x_n(t)), \end{cases} \tag{5}$$

where

$$\begin{cases} \hat{X}_{\gamma_1} \triangleq \{x_1, x_2, \ldots, x_n\} \backslash \{x_{\nu_1^1}, x_{\nu_1^2}, \ldots, x_{\nu_1^{\varepsilon_1}}\}, \\ \hat{X}_{\gamma_2} \triangleq \{x_1, x_2, \ldots, x_n\} \backslash \{x_{\nu_2^1}, x_{\nu_2^2}, \ldots, x_{\nu_2^{\varepsilon_2}}\}, \\ \quad\vdots \\ \hat{X}_{\gamma_\tau} \triangleq \{x_1, x_2, \ldots, x_n\} \backslash \{x_{\nu_\tau^1}, x_{\nu_\tau^2}, \ldots, x_{\nu_\tau^{\varepsilon_\tau}}\}. \end{cases}$$

Let $\hat{\boldsymbol{F}}_{\gamma_1} \in \boldsymbol{\mathcal{L}}_{2\times 2^{|\boldsymbol{N}_{\gamma_1}|}}$, $\hat{\boldsymbol{F}}_{\gamma_2} \in \boldsymbol{\mathcal{L}}_{2\times 2^{|\boldsymbol{N}_{\gamma_2}|}}$, ..., $\hat{\boldsymbol{F}}_{\gamma_\tau} \in \boldsymbol{\mathcal{L}}_{2\times 2^{|\boldsymbol{N}_{\gamma_\tau}|}}$ denote the corresponding reduced structure matrices of logical functions

$$\begin{cases} \hat{f}_{\gamma_1}\left(\hat{X}_{\gamma_1}(t)\right) : \mathcal{D}^{n-\varepsilon_1} \to \mathcal{D}, \\ \hat{f}_{\gamma_2}\left(\hat{X}_{\gamma_2}(t)\right) : \mathcal{D}^{n-\varepsilon_2} \to \mathcal{D}, \\ \vdots \\ \hat{f}_{\gamma_\tau}\left(\hat{X}_{\gamma_\tau}(t)\right) : \mathcal{D}^{n-\varepsilon_\tau} \to \mathcal{D}. \end{cases}$$

We assume that if $\hat{X}_{\gamma_i} = \varnothing$ $(i = 1, 2, \ldots, \tau)$, set $\hat{f}_{\gamma_i}\left(\hat{X}_{\gamma_i}(t)\right) \triangleq 1$ (or 0).

To further determine the logical functions of output feedback stabilizers, one needs to transform the structure matrices of nodes x_i $(i \in \Sigma_\tau)$. Here, denote $\boldsymbol{N}_{\gamma_1} \triangleq \boldsymbol{N}\backslash\boldsymbol{N}^{\rm c}_{\gamma_1}$, $\boldsymbol{N}_{\gamma_2} \triangleq \boldsymbol{N}\backslash\boldsymbol{N}^{\rm c}_{\gamma_2}$, ..., $\boldsymbol{N}_{\gamma_\tau} \triangleq \boldsymbol{N}\backslash\boldsymbol{N}^{\rm c}_{\gamma_\tau}$, $\boldsymbol{N}^{\rm c}_{\gamma_1} \triangleq \{\nu_1^1, \nu_1^2, \ldots, \nu_1^{\varepsilon_1}\}$, $\boldsymbol{N}^{\rm c}_{\gamma_2} \triangleq \{\nu_2^1, \nu_2^2, \ldots, \nu_2^{\varepsilon_2}\}$, ..., $\boldsymbol{N}^{\rm c}_{\gamma_\tau} \triangleq \{\nu_\varepsilon^1, \nu_\varepsilon^2, \ldots, \nu_\varepsilon^{\varepsilon_\tau}\}$. According to Eq. (5) with the reduced matrices $\hat{\boldsymbol{F}}_{\gamma_1} \in \boldsymbol{\mathcal{L}}_{2\times 2^{|\boldsymbol{N}_{\gamma_1}|}}$, $\hat{\boldsymbol{F}}_{\gamma_2} \in \boldsymbol{\mathcal{L}}_{2\times 2^{|\boldsymbol{N}_{\gamma_2}|}}$, ..., $\hat{\boldsymbol{F}}_{\gamma_\tau} \in \boldsymbol{\mathcal{L}}_{2\times 2^{|\boldsymbol{N}_{\gamma_\tau}|}}$, transform matrices $\hat{\boldsymbol{F}}_i$ $(i \in \Sigma_\tau)$ into

$$\tilde{\boldsymbol{F}}_i = \hat{\boldsymbol{F}}_i\left(\boldsymbol{1}^{\rm T}_{2^{|\boldsymbol{N}_i^{\rm c}|}} \otimes \boldsymbol{I}_{2^{|\boldsymbol{N}_i|}}\right), \quad i \in \Sigma_\tau. \tag{6}$$

Swap matrix $\boldsymbol{W}_{[m,n]}$ has the property $\boldsymbol{W}_{[m,n]}\sigma_1\sigma_2 = \sigma_2\sigma_1$. Then one has the following lemma:

Lemma 1 Considering $x_j(t+1) = \boldsymbol{F}_j \ltimes_{i=1}^n x_i(t)$ $(j \in \Sigma_\tau,\ i \in [1, \tau])$, one can swap the position of neighbors in the following form:

$$\begin{aligned} \boldsymbol{x}_{\gamma_i}(t+1) = &\ \boldsymbol{F}_{\gamma_i}\boldsymbol{W}_i \ltimes_{j\in\boldsymbol{N}^{\rm c}_{\gamma_i}} \boldsymbol{x}_j(t) \ltimes_{j\in\boldsymbol{N}_{\gamma_i}} \boldsymbol{x}_j(t) \\ \triangleq &\ \check{\boldsymbol{F}}_{\gamma_i} \ltimes_{j\in\boldsymbol{N}^{\rm c}_{\gamma_i}} \boldsymbol{x}_j(t) \ltimes_{j\in\boldsymbol{N}_{\gamma_i}} \boldsymbol{x}_j(t), \end{aligned} \tag{7}$$

where $\boldsymbol{W}_1 \triangleq \ltimes_{i=1}^{\varepsilon_1}\boldsymbol{W}_{[2,2^{\nu_1^i-1}]}$, $\boldsymbol{W}_2 \triangleq \ltimes_{i=1}^{\varepsilon_2}\boldsymbol{W}_{[2,2^{\nu_2^i-1}]}$, ..., $\boldsymbol{W}_\tau \triangleq \ltimes_{i=1}^{\varepsilon_\tau}\boldsymbol{W}_{[2,2^{\nu_\tau^i-1}]}$ are orthogonal matrices, and $\check{\boldsymbol{F}}_{\gamma_1} \triangleq \boldsymbol{F}_{\gamma_1}\boldsymbol{W}_1 \in \boldsymbol{\mathcal{L}}_{2\times 2^n}$, $\check{\boldsymbol{F}}_{\gamma_2} \triangleq \boldsymbol{F}_{\gamma_2}\boldsymbol{W}_2 \in \boldsymbol{\mathcal{L}}_{2\times 2^n}$, ..., $\check{\boldsymbol{F}}_{\gamma_\tau} \triangleq \boldsymbol{F}_{\gamma_\tau}\boldsymbol{W}_\tau \in \boldsymbol{\mathcal{L}}_{2\times 2^n}$.

Then one can design the following output feedback pinning control on nodes x_i $(i \in \Sigma_\tau)$:

$$\begin{cases} x_i(t+1) = u_i(t) \oplus_i f_i\left(x_1(t), x_2(t), \ldots, x_n(t)\right), \\ \qquad\qquad i \in \Sigma_\tau, \\ x_j(t+1) = f_j\left(x_1(t), x_2(t), \ldots, x_n(t)\right), \\ \qquad\qquad j \in N\backslash\Sigma_\tau, \\ u_i(t) = h_i\left(y_1(t), y_2(t), \ldots, y_p(t)\right), i \in \Sigma_\tau, \end{cases} \tag{8}$$

where $\oplus_i : \mathcal{D}^2 \to \mathcal{D}$ and $h_i : \mathcal{D}^p \to \mathcal{D}$ are logical functions to be determined. Denote $\boldsymbol{M}_i \in \boldsymbol{\mathcal{L}}_{2\times 4}$ and $\boldsymbol{H}_i \in \boldsymbol{\mathcal{L}}_{2\times 2^p}$ as corresponding structure matrices of functions $\oplus_i$ and h_i, which will be determined in the following analysis.

Based on matrices $\boldsymbol{W}_1, \boldsymbol{W}_2, \ldots, \boldsymbol{W}_\tau$, one can convert $\boldsymbol{y}(t) = \boldsymbol{G}\boldsymbol{x}(t)$ into

$$\begin{cases} \boldsymbol{y}_1(t) = \boldsymbol{G}_1 \ltimes_{j\in\boldsymbol{N}^{\rm c}_{\gamma_1}} \boldsymbol{x}_j(t) \ltimes_{j\in\boldsymbol{N}_{\gamma_1}} \boldsymbol{x}_j(t), \\ \boldsymbol{y}_2(t) = \boldsymbol{G}_2 \ltimes_{j\in\boldsymbol{N}^{\rm c}_{\gamma_2}} \boldsymbol{x}_j(t) \ltimes_{j\in\boldsymbol{N}_{\gamma_2}} \boldsymbol{x}_j(t), \\ \qquad \vdots \\ \boldsymbol{y}_\tau(t) = \boldsymbol{G}_\tau \ltimes_{j\in\boldsymbol{N}^{\rm c}_{\gamma_\tau}} \boldsymbol{x}_j(t) \ltimes_{j\in\boldsymbol{N}_{\gamma_\tau}} \boldsymbol{x}_j(t), \end{cases}$$

where $\boldsymbol{G}_1 \triangleq \boldsymbol{G}\boldsymbol{W}_1 \in \boldsymbol{\mathcal{L}}_{2^p\times 2^n}$, $\boldsymbol{G}_2 \triangleq \boldsymbol{G}\boldsymbol{W}_2 \in \boldsymbol{\mathcal{L}}_{2^p\times 2^n}$, ..., $\boldsymbol{G}_\tau \triangleq \boldsymbol{G}\boldsymbol{W}_\tau \in \boldsymbol{\mathcal{L}}_{2^p\times 2^n}$. Then under the output feedback stabilizer system $\boldsymbol{u}_i(t) = \boldsymbol{H}_i\boldsymbol{y}(t)$, one can obtain that for $i \in \Sigma_\tau$,

$$\begin{aligned} \boldsymbol{x}_i(t+1) &= \boldsymbol{u}_i(t) \oplus_i f_i\left(x_1(t), x_2(t), \ldots, x_n(t)\right) \\ &= \boldsymbol{M}_{\oplus_i}\boldsymbol{u}_i(t)\check{\boldsymbol{F}}_i \ltimes_{j\in\boldsymbol{N}^{\rm c}_{\gamma_i}} \boldsymbol{x}_j(t) \ltimes_{j\in\boldsymbol{N}_{\gamma_i}} \boldsymbol{x}_j(t) \\ &= \boldsymbol{M}_{\oplus_i}\boldsymbol{H}_i\boldsymbol{y}(t)\check{\boldsymbol{F}}_i \ltimes_{j\in\boldsymbol{N}^{\rm c}_{\gamma_i}} \boldsymbol{x}_j(t) \ltimes_{j\in\boldsymbol{N}_{\gamma_i}} \boldsymbol{x}_j(t) \\ &= \boldsymbol{M}_{\oplus_i}\boldsymbol{H}_i\boldsymbol{G}_i \ltimes_{j\in\boldsymbol{N}^{\rm c}_{\gamma_i}} \boldsymbol{x}_j(t) \ltimes_{j\in\boldsymbol{N}_{\gamma_i}} \boldsymbol{x}_j(t)\check{\boldsymbol{F}}_i \\ &\quad \ltimes_{j\in\boldsymbol{N}^{\rm c}_{\gamma_i}} \boldsymbol{x}_j(t) \ltimes_{j\in\boldsymbol{N}_{\gamma_i}} \boldsymbol{x}_j(t) \\ &= \boldsymbol{\varXi}_i \ltimes_{j\in\boldsymbol{N}^{\rm c}_{\gamma_i}} \boldsymbol{x}_j(t) \ltimes_{j\in\boldsymbol{N}_{\gamma_i}} \boldsymbol{x}_j(t). \end{aligned}$$

Denote $\boldsymbol{\varXi}_i = \boldsymbol{M}_{\oplus_i}\boldsymbol{H}_i\boldsymbol{G}_i(\boldsymbol{I}_{2^n} \otimes \check{\boldsymbol{F}}_i)\boldsymbol{\varPhi}_{2^n}$. Here, $\boldsymbol{M}_{\oplus_i} \in \boldsymbol{\mathcal{L}}_{2\times 4}$ and $\boldsymbol{H}_i \in \boldsymbol{\mathcal{L}}_{2\times 2^p}$ are structure matrices of functions $\oplus_i : \mathcal{D}^2 \to \mathcal{D}$ and $h_i : \mathcal{D}^p \to \mathcal{D}$ $(i \in \Sigma_\tau)$, respectively. These matrices $\boldsymbol{M}_{\oplus_i} \in \boldsymbol{\mathcal{L}}_{2\times 4}$ and $\boldsymbol{H}_i \in \boldsymbol{\mathcal{L}}_{2\times 2^p}$ will be determined in the following analysis.

Theorem 2 For each $i \in \Sigma_\tau$, if there exist matrices $\boldsymbol{M}_{\oplus_i}$ and $\boldsymbol{H}_i$ such that $\boldsymbol{M}_{\oplus_i} \ltimes \boldsymbol{H}_i \ltimes \boldsymbol{G}_i \ltimes (\boldsymbol{I}_{2^n} \otimes \check{\boldsymbol{F}}_i) \ltimes \boldsymbol{\varPhi}_{2^n} = \tilde{\boldsymbol{F}}_i$ and $\tilde{\boldsymbol{F}}_i = \hat{\boldsymbol{F}}_i(\boldsymbol{1}^{\rm T}_{2^{|\boldsymbol{N}_i^{\rm c}|}} \otimes \boldsymbol{I}_{2^{|\boldsymbol{N}_i|}})$ hold, where $\hat{\boldsymbol{F}}_i$ and $\tilde{\boldsymbol{F}}_i$ are matrices obtained in Eqs. (5) and (6) respectively, then the wiring digraph $G = (V, E)$ will be acyclic.

Proof One can obtain that

$$\begin{aligned} \boldsymbol{x}_i(t+1) =& \boldsymbol{M}_{\oplus_i}\boldsymbol{H}_i\boldsymbol{G}_i(\boldsymbol{I}_{2^n} \otimes \check{\boldsymbol{F}}_i)\boldsymbol{\varPhi}_{2^n} \\ & \ltimes_{j\in\boldsymbol{N}^{\rm c}_{\gamma_i}} \boldsymbol{x}_j(t) \ltimes_{j\in\boldsymbol{N}_{\gamma_i}} \boldsymbol{x}_j(t) \\ =& \tilde{\boldsymbol{F}}_i \ltimes_{j\in\boldsymbol{N}^{\rm c}_{\gamma_i}} \boldsymbol{x}_j(t) \ltimes_{j\in\boldsymbol{N}_{\gamma_i}} \boldsymbol{x}_j(t) \\ =& \hat{\boldsymbol{F}}_i(\boldsymbol{1}^{\rm T}_{2^{|\boldsymbol{N}_i^{\rm c}|}} \otimes I_{2^{|\boldsymbol{N}_i|}}) \\ & \ltimes_{j\in\boldsymbol{N}^{\rm c}_{\gamma_i}} \boldsymbol{x}_j(t) \ltimes_{j\in\boldsymbol{N}_{\gamma_i}} \boldsymbol{x}_j(t) \\ =& \hat{\boldsymbol{F}}_i \ltimes_{j\in\boldsymbol{N}_{\gamma_i}} \boldsymbol{x}_j(t). \end{aligned}$$

Thus, for nodes x_i $(i \in \Sigma_\tau)$ in the wiring digraph $G = (V, E)$, the corresponding edges $e_1, e_2, \ldots, e_p \in E$ will be deleted. Since the edges $e_1, e_2, \ldots, e_p \in E$ are a feasible FAS, the deleted edges will lead to the wiring digraph $G=(V,E)$ being acyclic, which guarantees stability.

Thus, the solvability of unknown matrices $\boldsymbol{M}_{\oplus_i}$ and $\boldsymbol{H}_i$ of equations $\boldsymbol{M}_{\oplus_i}\boldsymbol{H}_i\boldsymbol{G}_i(\boldsymbol{I}_{2^n}\otimes\check{\boldsymbol{F}}_i)\varPhi_{2^n}=\tilde{\boldsymbol{F}}_i$ ($i\in\varSigma_\tau$) plays a vital role in the design of the output feedback stabilizer. Denote

$$\begin{cases}\hat{\boldsymbol{H}}_{\gamma_1}=\boldsymbol{H}_{\gamma_1}\boldsymbol{G}_i\in\boldsymbol{\mathcal{L}}_{2\times 2^n},\\ \hat{\boldsymbol{H}}_{\gamma_2}=\boldsymbol{H}_{\gamma_2}\boldsymbol{G}_i\in\boldsymbol{\mathcal{L}}_{2\times 2^n},\\ \qquad\vdots\\ \hat{\boldsymbol{H}}_{\gamma_\tau}=\boldsymbol{H}_{\gamma_\tau}\boldsymbol{G}_i\in\boldsymbol{\mathcal{L}}_{2\times 2^n}.\end{cases}$$

Then as for the controlled equations $x_i(t+1)=u_i(t)\oplus_i f_i(x_1(t),\ldots,x_n(t))$ and $u_i(t)=h_i(y_1(t),\ldots,y_p(t))$ ($i\in\varSigma_\tau$), one has the following equations:

$$\boldsymbol{M}_{\oplus_i}\hat{\boldsymbol{H}}_i(\boldsymbol{I}_{2^n}\otimes\check{\boldsymbol{F}}_i)\boldsymbol{\varPhi}_{2^n}=\tilde{\boldsymbol{F}}_i,\quad i\in\varSigma_\tau,\tag{9a}$$

$$\hat{\boldsymbol{H}}_i=\boldsymbol{H}_i\boldsymbol{G}_i,\quad i\in\varSigma_\tau.\tag{9b}$$

Here, $\boldsymbol{M}_{\oplus_i}\in\boldsymbol{\mathcal{L}}_{2\times 4}$, $\boldsymbol{H}_i\in\boldsymbol{\mathcal{L}}_{2\times 2^p}$, and $\hat{\boldsymbol{H}}_i\in\boldsymbol{\mathcal{L}}_{2\times 2^n}$ are to be solved. In Li (2015), state feedback pinning controllers were first studied, and Eq. (9a) was proved to be solvable to obtain solutions for matrices $\boldsymbol{M}_{\oplus_i}$ and $\hat{\boldsymbol{H}}_i$ ($i\in\varSigma_\tau$). The detailed proof for the solvability of Eq. (9a) can be found in Li (2015).

Since Eq. (9a) is solvable for solutions for matrices $\boldsymbol{M}_{\oplus_i}\in\boldsymbol{\mathcal{L}}_{2\times 4}$ and $\hat{\boldsymbol{H}}_i\in\boldsymbol{\mathcal{L}}_{2\times 2^n}$ ($i\in\varSigma_\tau$), we assume that for each $i\in\varSigma_\tau$, $\varTheta_i=\{(\boldsymbol{M}_{\oplus_i},\hat{\boldsymbol{H}}_i):\boldsymbol{M}_{\oplus_i},\hat{\boldsymbol{H}}_i$ satisfies Eq. (9a)$\}$ are all the feasible solutions for Eq. (9a). For each $i\in\varSigma_\tau$, we set

$$\varOmega_i=\{\hat{\boldsymbol{H}}_i:\hat{\boldsymbol{H}}_i\text{ satisfies Eq. (9a)}\}.\tag{10}$$

For each $i\in\varSigma_\tau$, denote $\boldsymbol{G}_i=\boldsymbol{\delta}_{2^p}[g_1^i,g_2^i,\ldots,g_{2^n}^i]$, where $g_1^i,g_2^i,\ldots,g_{2^n}^i\in[1,2^p]$. Then for $\vartheta\in[1,2^p]$, define $\varSigma_i(\vartheta)=\{\boldsymbol{\delta}_{2^n}^j:\mathrm{Col}_j(\boldsymbol{G}_i)=\boldsymbol{\delta}_{2^p}^{\vartheta},\ j\in[1,2^n]\}$, where $i\in\varSigma_\tau$. Obviously, $\varSigma_i(\vartheta_1)\cap\varSigma_i(\vartheta_2)=\varnothing$ ($\forall\vartheta_1\neq\vartheta_2$), and $\bigcup_{\vartheta=1}^{2^n}\varSigma_i(\vartheta)=\varDelta_{2^n}$ ($i\in\varSigma_\tau$).

Theorem 3 (Solvability of Eqs. (9a) and (9b)) Consider the solvability of Eqs. (9a) and (9b). They are solvable if and only if for every $i\in\varSigma_\tau$ with corresponding set $\varOmega_i$, there exists a matrix $\hat{\boldsymbol{H}}_i\in\varOmega_i$ such that

$$\mathrm{Col}_{j_1}(\hat{\boldsymbol{H}}_i)=\mathrm{Col}_{j_2}(\hat{\boldsymbol{H}}_i),\forall j_1,j_2\in\varSigma_i(\vartheta),\forall\vartheta\in[1,2^p].\tag{11}$$

Proof First, let us prove the necessity. Suppose that Eqs. (9a) and (9b) are solvable for each $i\in\varSigma_\tau$. Due to the solvability of Eq. (9a), one can obtain all the feasible solutions for matrices $\boldsymbol{M}_{\oplus_i}\in\boldsymbol{\mathcal{L}}_{2\times 4}$ and $\hat{\boldsymbol{H}}_i\in\boldsymbol{\mathcal{L}}_{2\times 2^n}$ ($i\in\varSigma_\tau$). Then for each $i\in\varSigma_\tau$, one can obtain the set $\varOmega_i=\{\hat{\boldsymbol{H}}_i:\hat{\boldsymbol{H}}_i$ satisfies Eq. (9a)$\}$. Fixing $i\in\varSigma_\tau$, choose a feasible solution $\boldsymbol{X}\in\mathcal{L}_{2\times 4}$, $\boldsymbol{Y}\in\varOmega_i\in\boldsymbol{\mathcal{L}}_{2\times 2^n}$, $\boldsymbol{Z}\in\boldsymbol{\mathcal{L}}_{2\times 2^p}$, such that the following equations hold:

$$\begin{cases}\boldsymbol{X}\ltimes\boldsymbol{Y}\ltimes(\boldsymbol{I}_{2^n}\otimes\check{\boldsymbol{F}}_i)\ltimes\boldsymbol{\varPhi}_{2^n}=\tilde{\boldsymbol{F}}_i,\\ \boldsymbol{Y}=\boldsymbol{Z}\boldsymbol{G}_i.\end{cases}\tag{12}$$

Eq. (12) implies that for each $\boldsymbol{\alpha}=\boldsymbol{\delta}_{2^n}^j$, $j\in[1,2^n]$, one has $\mathrm{Col}_j(\boldsymbol{Y})=\boldsymbol{Z}\mathrm{Col}_j(\boldsymbol{G}_i)$. Then if $\mathrm{Col}_j(\boldsymbol{G}_i)=\boldsymbol{\delta}_{2^p}^{\vartheta}$, we have $\boldsymbol{\alpha}=\boldsymbol{\delta}_{2^n}^j\in\varSigma_i(\vartheta)$ and $\mathrm{Col}_\vartheta(\boldsymbol{Z})=\mathrm{Col}_j(\boldsymbol{Y})$. Thus, for any pair of distinct $\boldsymbol{\kappa}=\boldsymbol{\delta}_{2^n}^{\mu}$, $\boldsymbol{\beta}=\boldsymbol{\delta}_{2^n}^{\nu}\in\varSigma_i(\vartheta)$ ($\mu\neq\nu\in[1,2^n]$), one should obtain $\mathrm{Col}_\vartheta(\boldsymbol{Z})=\mathrm{Col}_\mu(\boldsymbol{Y})=\mathrm{Col}_\nu(\boldsymbol{Y})$. Since $i\in\varSigma_\tau$ is arbitrarily given, the proof of necessity is completed. The proof of sufficiency can be directly obtained, which is omitted here. The proof is completed.

Based on the above analysis, one can design an output feedback stabilizer to achieve a unique steady state. Among all the feasible solutions $\varTheta_i=\{(\boldsymbol{M}_{\oplus_i},\hat{\boldsymbol{H}}_i):\boldsymbol{M}_{\oplus_i},\hat{\boldsymbol{H}}_i$ satisfies Eq. (9a)$\}$, $i\in\varSigma_\tau$, find a feasible matrix $\hat{\boldsymbol{H}}_i^{\circ}$ satisfying condition (11). Then calculate the output feedback matrices. For each $i\in\varSigma_\tau$ and $\vartheta\in[1,2^p]$, calculate matrix $\boldsymbol{H}_i$, as follows:

$$\mathrm{Col}_\vartheta(\boldsymbol{H}_i)=\begin{cases}\mathrm{Col}_s(\hat{\boldsymbol{H}}_i), & \boldsymbol{\delta}_{2^n}^s\in\varSigma_i(\vartheta)\neq\varnothing,\\ \{\boldsymbol{\delta}_2^1,\boldsymbol{\delta}_2^2\}, & \varSigma_i(\vartheta)=\varnothing.\end{cases}\tag{13}$$

Finally, one can design the following output feedback stabilizer to achieve a unique steady state, where the controllers are imposed on nodes x_i ($i\in\varSigma_\tau$):

$$\begin{cases}\boldsymbol{x}_i(t+1)=\boldsymbol{u}_i(t)\oplus_i f_i\Big(\boldsymbol{x}_1(t),\boldsymbol{x}_2(t),\ldots,\boldsymbol{x}_n(t)\Big),\\ \qquad\qquad\qquad\qquad i\in\varSigma_\tau,\\ \boldsymbol{x}_j(t+1)=f_j\Big(\boldsymbol{x}_1(t),\boldsymbol{x}_2(t),\ldots,\boldsymbol{x}_n(t)\Big),\\ \qquad\qquad\qquad\qquad j\in N\backslash\varSigma_\tau,\\ \boldsymbol{u}_i(t)=h_i\Big(\boldsymbol{y}_1(t),\boldsymbol{y}_2(t),\ldots,\boldsymbol{y}_p(t)\Big),i\in\varSigma_\tau.\end{cases}\tag{14}$$

Here, the corresponding matrices $\boldsymbol{H}_{\gamma_1},\boldsymbol{H}_{\gamma_2},\ldots,\boldsymbol{H}_{\gamma_\tau}$ satisfying condition (13) are structure matrices for logical functions $h_{\gamma_1},h_{\gamma_2},\ldots,h_{\gamma_\tau}:\mathcal{D}^p\to\mathcal{D}$, and matrices $\boldsymbol{M}_{\gamma_1}^{\text{o}}, \boldsymbol{M}_{\gamma_2}^{\text{o}}, \ldots, \boldsymbol{M}_{\gamma_\tau}^{\text{o}}$ together with matrices $\hat{\boldsymbol{H}}_{\gamma_1}^{\text{o}}, \hat{\boldsymbol{H}}_{\gamma_2}^{\text{o}}, \ldots, \hat{\boldsymbol{H}}_{\gamma_\tau}^{\text{o}}$ are feasible solutions for Eqs. (9a) and (9b).

To sum up, a detailed algorithm for designing an output feedback stabilizer to achieve global stabilization can be summarized in Algorithm 1.

Algorithm 1 Summarized procedures for designing an output feedback stabilizer for system (1)

1: Use the depth-first search algorithm to find a possible FAS, $\{e_1, e_2, \ldots, e_p\}$, the deletion of which will lead to the wiring digraph of system (1) being acyclic
2: Let $\bigcup_{j=1}^{p} \boldsymbol{P}_{+}(e_j) = \Sigma_\tau$. Consider nodes x_i $(i \in \Sigma_\tau)$, transform functions by Eq. (5), and obtain matrices $\hat{\boldsymbol{F}}_{\gamma_1} \in \boldsymbol{\mathcal{L}}_{2\times 2^{|\boldsymbol{N}_{\gamma_1}|}}$, $\hat{\boldsymbol{F}}_{\gamma_2} \in \boldsymbol{\mathcal{L}}_{2\times 2^{|\boldsymbol{N}_{\gamma_2}|}}$, …, $\hat{\boldsymbol{F}}_{\gamma_\tau} \in \boldsymbol{\mathcal{L}}_{2\times 2^{|\boldsymbol{N}_{\gamma_\tau}|}}$
3: Transform matrices $\boldsymbol{F}_i$ $(i \in \Sigma_\tau)$ according to Eq. (6)
4: Solve matrices $\boldsymbol{M}_{\oplus_i} \in \boldsymbol{\mathcal{L}}_{2\times 4}$ and $\hat{\boldsymbol{H}}_i \in \boldsymbol{\mathcal{L}}_{2\times 2^n}$ $(i \in \Sigma_\tau)$ from Eqs. (9a) and (9b), and obtain set Ω_i given by Eq. (10)
5: Based on Theorem 3, find solvable matrices $\hat{\boldsymbol{H}}_i^{\text{o}} \in \Omega_i$ satisfying Eq. (11), together with $\boldsymbol{M}_{\oplus_i}^{\text{o}}$ holds for Eqs. (9a) and (9b), $i \in \Sigma_\tau$; for each $i \in \Sigma_\tau$, calculate matrices $\boldsymbol{H}_{\gamma_1}$, $\boldsymbol{H}_{\gamma_2}$, …, $\boldsymbol{H}_{\gamma_\tau}$ from Eq. (13)
6: Finally, design an output feedback stabilizer on nodes x_i $(i \in \Sigma_\tau)$ in the form of Eq. (8), and the structure matrices for functions $\oplus_i : \mathcal{D}^2 \to \mathcal{D}$, $h_i : \mathcal{D}^p \to \mathcal{D}$ $(i \in \Sigma_\tau)$ are given by $\hat{\boldsymbol{H}}_i^{\text{o}}$ and $\boldsymbol{M}_{\oplus_i}^{\text{o}}$ $(i \in \Sigma_\tau)$

Remark 2 Compared with Li and Wang (2013) and Bof et al. (2015), in this study, the information of state transition matrix $\boldsymbol{L}$ is not used. This will reduce the computational complexity from $O(2^n \times 2^n)$ to $O(2 \times 2^n)$, where n is the number of nodes in BNs. Actually, the total computational complexity proposed in this study is divided into two parts: the first part is polynomial with the number of nodes in BNs using NS of BNs, and the second part is exponential with the number of nodes in BNs.

Remark 3 Compared with the traditional analysis on BNs using information of the state transition matrix, in this study, network structure of BNs and each corresponding algebraic form of nodes are used to design an output feedback pinning control. Thus, the proposed method can be applied to some large-scale networks. One interesting topic is how to combine the information of the state transition matrix and network structure of BNs to design a more efficient pinning control technique.

4 Simulations

In this section, a reduced signal transduction network and a reduced D. melanogaster segmentation polarity gene network are presented to demonstrate the validity of the theoretical results obtained.

Example 1 Consider a reduced signal transduction network (Saadatpour et al., 2010) consisting of three nodes, i.e., CIS, Ca_c^{2+}, and Ca^{2+}ATPase. In this network, we use $\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3$ to denote CIS, Ca_c^{2+}, Ca^{2+}ATPase, respectively. The logical evolution dynamic is given as follows:

$$\begin{cases} \boldsymbol{x}_1(t+1) = \neg \boldsymbol{x}_2(t), \\ \boldsymbol{x}_2(t+1) = \boldsymbol{x}_1(t) \wedge \neg \boldsymbol{x}_3(t), \\ \boldsymbol{x}_3(t+1) = \boldsymbol{x}_2(t). \end{cases} \tag{15}$$

We are interested in the activity levels of CIS and Ca_c^{2+}, with the output system given as follows:

$$\begin{cases} \boldsymbol{y}_1(t) = \boldsymbol{x}_1(t), \\ \boldsymbol{y}_2(t) = \boldsymbol{x}_3(t). \end{cases}$$

Denote $\boldsymbol{x}(t) = \ltimes_{i=1}^{3} \boldsymbol{x}_i(t)$ and $\boldsymbol{y}(t) = \boldsymbol{y}_1(t) \ltimes \boldsymbol{y}_2(t)$. Using the STP method, one has the algebraic forms:

$$\begin{cases} \boldsymbol{x}_1(t+1) = \boldsymbol{F}_1 \boldsymbol{x}(t), \\ \boldsymbol{x}_2(t+1) = \boldsymbol{F}_2 \boldsymbol{x}(t), \\ \boldsymbol{x}_3(t+1) = \boldsymbol{F}_3 \boldsymbol{x}(t), \end{cases}$$

where

$$\begin{cases} \boldsymbol{F}_1 = \boldsymbol{\delta}_2[2, 2, 1, 1, 2, 2, 1, 1], \\ \boldsymbol{F}_2 = \boldsymbol{\delta}_2[2, 1, 2, 1, 2, 2, 2, 2], \\ \boldsymbol{F}_3 = \boldsymbol{\delta}_2[1, 1, 2, 2, 1, 1, 2, 2]. \end{cases}$$

Then we have the following algebraic forms:

$$\begin{cases} \boldsymbol{x}(t+1) = \boldsymbol{L}\boldsymbol{x}(t), \\ \boldsymbol{y}(t) = \boldsymbol{G}\boldsymbol{x}(t), \end{cases}$$

where

$$\begin{cases} \boldsymbol{L} = \boldsymbol{\delta}_8[7, 5, 4, 2, 7, 7, 4, 4], \\ \boldsymbol{G} = \boldsymbol{\delta}_4[1, 2, 1, 2, 3, 4, 3, 4]. \end{cases}$$

Fig. 1a shows the wiring digraph $G = (V, E)$ of Eq. (15). Actually, by calculations of state transition matrix $\boldsymbol{L}$, one can obtain that system (15) is not globally stable.

In the wiring digraph of system (15) shown in Fig. 1a, there are two cycles, i.e., $\{1, 2\}$ and $\{2, 3\}$.

The deletions of edges $1 \to 2$ and $2 \to 3$ will lead the NS of system (15) to be acyclic. Thus, we will design an output feedback stabilizer on nodes x_2 and x_3. First, according to Eq. (6), find matrices

$$\begin{cases} \tilde{\boldsymbol{F}}_2 = \boldsymbol{\delta}_2[2,1,2,1,2,1,2,1], \\ \tilde{\boldsymbol{F}}_3 = \boldsymbol{\delta}_2[1,1,1,1,1,1,1,1]. \end{cases}$$

Now, we will design two output feedback stabilizers $\boldsymbol{u}_2$ and $\boldsymbol{u}_3$ on nodes $\boldsymbol{x}_2$ and $\boldsymbol{x}_3$ respectively in the following form:

$$\begin{cases} \boldsymbol{x}_2(t+1) = \boldsymbol{u}_2(t) \oplus_2 [\boldsymbol{x}_1(t) \wedge \neg \boldsymbol{x}_3(t)], \\ \boldsymbol{u}_2(t) = h_2(\boldsymbol{y}_1(t), \boldsymbol{y}_2(t)), \\ \boldsymbol{x}_3(t+1) = \boldsymbol{u}_3(t) \oplus_3 [\boldsymbol{x}_2(t)], \\ \boldsymbol{u}_3(t) = h_3(\boldsymbol{y}_1(t), \boldsymbol{y}_2(t)). \end{cases} \tag{16}$$

Here $\boldsymbol{M}_2, \boldsymbol{M}_3 \in \boldsymbol{\mathcal{L}}_{2\times 4}$, $\boldsymbol{H}_2, \boldsymbol{H}_3 \in \boldsymbol{\mathcal{L}}_{2\times 4}$ are structure matrices of logical functions $\oplus_2, \oplus_3 : \mathcal{D}^2 \to \mathcal{D}$, $h_2, h_3 : \mathcal{D}^2 \to \mathcal{D}$, which will be determined below.

Denote $\hat{\boldsymbol{H}}_2 = \boldsymbol{H}_2\boldsymbol{G}$ and $\hat{\boldsymbol{H}}_3 = \boldsymbol{H}_3\boldsymbol{G}$. Using the STP method, one first obtains the following equations:

$$\boldsymbol{M}_{\oplus_i} \ltimes \hat{\boldsymbol{H}}_i \ltimes (\boldsymbol{I}_{2^3} \otimes \boldsymbol{F}_i) \ltimes \boldsymbol{\Phi}_{2^3} = \tilde{\boldsymbol{F}}_i, \tag{17a}$$

$$\hat{\boldsymbol{H}}_i = \boldsymbol{H}_i\boldsymbol{G}, \tag{17b}$$

where $i = 2, 3$.

By calculation, one can obtain that there are 12 feasible solutions for $\Theta_2 = \{(\boldsymbol{M}_{\oplus_2}, \hat{\boldsymbol{H}}_2) : \boldsymbol{M}_{\oplus_2}, \hat{\boldsymbol{H}}_2$ satisfies Eq. (17a)$\}$, and 324 feasible solutions for $\Theta_3 = \{(\boldsymbol{M}_{\oplus_3}, \hat{\boldsymbol{H}}_3) : \boldsymbol{M}_{\oplus_3}, \hat{\boldsymbol{H}}_3$ satisfies Eq. (17a)$\}$. Then one can find a feasible solution

$$\begin{cases} \boldsymbol{M}_2^{\circ} = \boldsymbol{\delta}_2[2,1,1,2], \\ \hat{\boldsymbol{H}}_2^{\circ} = \boldsymbol{\delta}_2[2,2,2,2,2,1,2,1], \\ \boldsymbol{M}_3^{\circ} = \boldsymbol{\delta}_2[1,1,1,2], \\ \hat{\boldsymbol{H}}_3^{\circ} = \boldsymbol{\delta}_2[1,1,1,1,1,1,1,1], \end{cases}$$

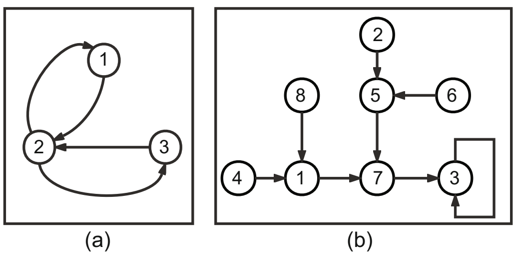

Fig. 1 Original wiring digraph $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E})$ of system (15) (a) and the state transition graph of system (15) under output feedback stabilizers (18) (b). The number i in each circle presents the state $\boldsymbol{\delta}_8^i$, $i = [1, 8]$

which satisfies Theorem 3. Thus, one can derive matrices $\boldsymbol{H}_2 = \boldsymbol{\delta}_2[2,2,2,1]$ and $\boldsymbol{H}_3 = \boldsymbol{\delta}_2[1,1,1,1]$. Then we can use the reconstruction theory in Cheng et al. (2010) to construct logical functions. Finally, one can obtain the logical dynamics of output feedback stabilizers as follows:

$$\begin{cases} \boldsymbol{x}_2(t+1) = [\boldsymbol{u}_2(t) \wedge \neg(\boldsymbol{x}_1(t) \wedge \neg \boldsymbol{x}_3(t))] \\ \quad \vee [\neg \boldsymbol{u}_2(t) \wedge (\boldsymbol{x}_1(t) \wedge \neg \boldsymbol{x}_3(t))], \\ \boldsymbol{u}_2(t) = \neg \boldsymbol{y}_1(t) \wedge \neg \boldsymbol{y}_2(t), \\ \boldsymbol{x}_3(t+1) = \boldsymbol{u}_3(t) \vee \boldsymbol{x}_2(t), \\ \boldsymbol{u}_3(t) = \mathbf{1}. \end{cases} \tag{18}$$

Then under the above output feedback stabilizers in the form of Eq. (18), system (15) will achieve global stabilization. Fig. 1b shows the state transition digraph of system (15) under the output feedback stabilizers (18). It also shows that system (15) under output feedback stabilizers in the form of Eq. (18) achieves global stabilization.

Example 2 Consider a reduced D. melanogaster segmentation polarity gene network introduced in Xiao and Dougherty (2007) and Li (2015), where the six nodes (genes) are denoted as wg1 = $\boldsymbol{x}_1$, wg2 = $\boldsymbol{x}_2$, wg3 = $\boldsymbol{x}_3$, wg4 = $\boldsymbol{x}_4$, PTC1 = $\boldsymbol{x}_5$, and PTC2 = $\boldsymbol{x}_6$. The D. melanogaster segmentation polarity gene network has been modeled as a BN in Li (2015). The logical dynamics are given in the following form:

$$\begin{cases} \boldsymbol{x}_1(t+1) = \neg \boldsymbol{x}_2(t) \wedge \neg \boldsymbol{x}_4(t), \\ \boldsymbol{x}_2(t+1) = \neg \boldsymbol{x}_1(t) \wedge \neg \boldsymbol{x}_3(t), \\ \boldsymbol{x}_3(t+1) = \boldsymbol{x}_1(t), \\ \boldsymbol{x}_4(t+1) = \boldsymbol{x}_2(t), \\ \boldsymbol{x}_5(t+1) = (\neg \boldsymbol{x}_2(t) \wedge \neg \boldsymbol{x}_4(t)) \\ \qquad \vee (\neg \boldsymbol{x}_1(t) \wedge \neg \boldsymbol{x}_3(t)), \\ \boldsymbol{x}_6(t+1) = (\neg \boldsymbol{x}_1(t) \wedge \neg \boldsymbol{x}_3(t)) \\ \qquad \vee (\neg \boldsymbol{x}_2(t) \wedge \neg \boldsymbol{x}_4(t)). \end{cases} \tag{19}$$

We are interested in the activity levels of gene structures wg2 and wg4, that is, nodes $\boldsymbol{x}_2$ and $\boldsymbol{x}_4$, with the output system given as follows: $\boldsymbol{y}_1(t) = \boldsymbol{x}_2(t)$, $\boldsymbol{y}_2(t) = \boldsymbol{x}_4(t)$. Denote $\boldsymbol{x}(t) = \ltimes_{i=1}^6 \boldsymbol{x}_i(t)$ and $\boldsymbol{y}(t) = \boldsymbol{y}_1(t) \ltimes \boldsymbol{y}_2(t)$. Using the STP method, one has the algebraic forms: $\boldsymbol{x}_i(t+1) = \boldsymbol{F}_i\boldsymbol{x}(t)$ $(i = 1, 2, \ldots, 6)$, where

$$\begin{cases} \boldsymbol{F}_1 = \boldsymbol{M}_{\rm c}\boldsymbol{M}_{\rm n}(\boldsymbol{I}_2\otimes\boldsymbol{M}_{\rm n})(\boldsymbol{I}_2\otimes[1,1])([1,1]\otimes_2)(\boldsymbol{I}_{2^4}\oplus\boldsymbol{1}_4), \\ \boldsymbol{F}_2 = \boldsymbol{M}_{\rm c}\boldsymbol{M}_{\rm n}(\boldsymbol{I}_2\otimes\boldsymbol{M}_{\rm n})(\boldsymbol{I}_2\otimes[1,1])(\boldsymbol{I}_8\otimes\boldsymbol{1}_8), \\ \boldsymbol{F}_3 = (\boldsymbol{I}_2\otimes\boldsymbol{1}_{2^5}), \\ \boldsymbol{F}_4 = ([1,1]\otimes\boldsymbol{I}_2)(\boldsymbol{I}_4\otimes\boldsymbol{1}_{2^4}), \\ \boldsymbol{F}_5 = \boldsymbol{M}_{\rm d}\boldsymbol{M}_{\rm c}\boldsymbol{M}_{\rm n}(\boldsymbol{I}_2\otimes\boldsymbol{M}_{\rm n})(\boldsymbol{I}_4\otimes(\boldsymbol{M}_{\rm c}\boldsymbol{M}_{\rm n}(\boldsymbol{I}_2 \\ \qquad \otimes\boldsymbol{M}_{\rm n})))\boldsymbol{W}_{[2,4]}(\boldsymbol{I}_4\otimes\boldsymbol{W}_{[2]})(I_{2^4}\otimes\boldsymbol{1}_{2^2}), \\ \boldsymbol{F}_6 = \boldsymbol{M}_{\rm d}\boldsymbol{M}_{\rm c}\boldsymbol{M}_{\rm n}(\boldsymbol{I}_2\otimes\boldsymbol{M}_{\rm n})(\boldsymbol{I}_4\otimes(\boldsymbol{M}_{\rm c}\boldsymbol{M}_{\rm n}(\boldsymbol{I}_2 \\ \qquad \otimes\boldsymbol{M}_{\rm n})))(I_2\otimes\boldsymbol{M}_{\rm n})(\boldsymbol{I}_{2^4}\otimes\boldsymbol{1}_4), \end{cases}$$

and the output system: $\boldsymbol{y}(t)=\boldsymbol{G}\boldsymbol{x}(t)$, $\boldsymbol{G}=(\boldsymbol{1}_2\otimes\boldsymbol{I}_2)(\boldsymbol{I}_4\otimes\boldsymbol{1}_2)(\boldsymbol{I}_{2^4}\otimes\boldsymbol{1}_4)$. By calculation, one can obtain that system (19) is not in a globally stable configuration. Fig. 2 shows the wiring digraph $G=(V,E)$ of system (19), and Fig. 3 shows the state transition digraph, which has three attractors (one cyclic attractor with length three and two steady states).

Now, we will design an output feedback stabilizer to achieve global stabilization based on Fig. 2. In Fig. 2, there exists one feasible FAS consisting of edges $\boldsymbol{x}_2\to\boldsymbol{x}_1$ and $\boldsymbol{x}_2\to\boldsymbol{x}_4$, the deletions of which will lead to Fig. 2 being acyclic. Thus, we will design an output feedback stabilizer on nodes $\boldsymbol{x}_1$ and $\boldsymbol{x}_4$. First, according to Eq. (6), find matrices $\tilde{\boldsymbol{F}}_1=\boldsymbol{M}_n(\boldsymbol{1}_8\otimes\boldsymbol{I}_2)(\boldsymbol{I}_{2^4}\otimes\boldsymbol{1}_4)$ and $\tilde{\boldsymbol{F}}_4=\boldsymbol{\delta}_2[\overbrace{1,1,\ldots,1}^{2^6}]$. Now, we will design two output feedback stabilizers $\boldsymbol{u}_1$ and $\boldsymbol{u}_4$ on nodes $\boldsymbol{x}_1$ and $\boldsymbol{x}_4$ respectively in the following form:

$$\begin{cases} \boldsymbol{x}_1(t+1)=\boldsymbol{u}_1(t)\oplus_1[\neg\boldsymbol{x}_2(t)\wedge\neg\boldsymbol{x}_4(t)], \\ \boldsymbol{u}_1(t)=h_1(\boldsymbol{y}_1(t),\boldsymbol{y}_2(t)), \\ \boldsymbol{x}_4(t+1)=\boldsymbol{u}_4(t)\oplus_4[\boldsymbol{x}_2(t)], \\ \boldsymbol{u}_4(t)=h_4(\boldsymbol{y}_1(t),\boldsymbol{y}_2(t)). \end{cases} \tag{20}$$

Here $\boldsymbol{M}_1,\boldsymbol{M}_4\in\boldsymbol{\mathcal{L}}_{2\times4}$, $\boldsymbol{H}_1,\boldsymbol{H}_4\in\boldsymbol{\mathcal{L}}_{2\times4}$ are structure matrices of logical functions $\oplus_1:\mathcal{D}^2\to\mathcal{D}$, $\oplus_4:\mathcal{D}^2\to\mathcal{D}$, $h_1,h_4:\mathcal{D}^2\to\mathcal{D}$, which will be determined below.

Denote $\hat{\boldsymbol{H}}_1=\boldsymbol{H}_1\boldsymbol{G}$ and $\hat{\boldsymbol{H}}_4=\boldsymbol{H}_4\boldsymbol{G}$. Using the STP method, one first obtains the following equations:

$$\boldsymbol{M}_{\oplus_i}\ltimes\hat{\boldsymbol{H}}_i\ltimes(\boldsymbol{I}_{2^6}\otimes\boldsymbol{F}_i)\ltimes\boldsymbol{\Phi}_{2^6}=\tilde{\boldsymbol{F}}_i, \tag{21a}$$

$$\hat{\boldsymbol{H}}_i=\boldsymbol{H}_i\boldsymbol{G}, \tag{21b}$$

where $i=1,4$.

By calculation, one can obtain one feasible solution

$$\begin{cases} \boldsymbol{M}_{\oplus_1}=\boldsymbol{\delta}_2[1,1,1,2], \\ \boldsymbol{H}_1=\boldsymbol{\delta}_2[2,1,2,1], \\ \boldsymbol{M}_{\oplus_4}=\boldsymbol{\delta}_2[1,2,2,2], \\ \boldsymbol{H}_4=\boldsymbol{\delta}_2[2,2,1,1], \end{cases}$$

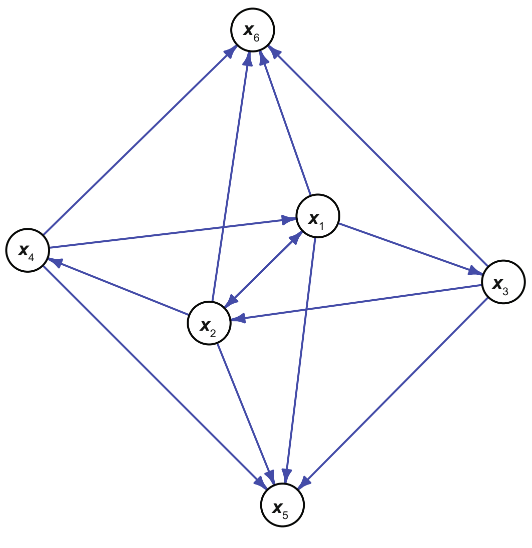

Fig. 2 Wiring digraph $G=(V,E)$ of system (19)

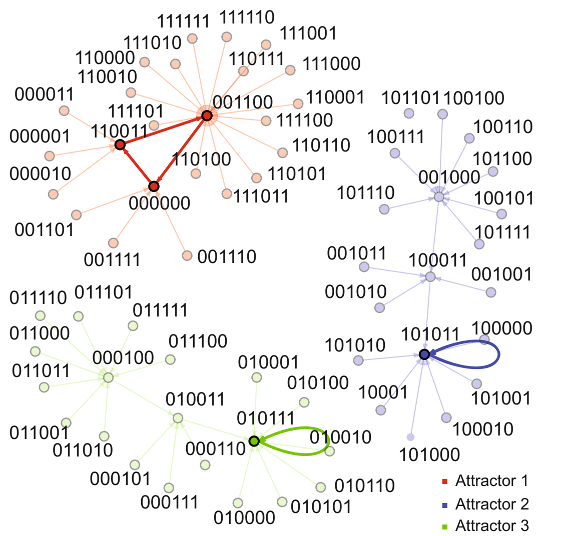

Fig. 3 State transition digraph of system (19), which has three attractors including two steady states and one cyclic attractor with length three. Each node represents a state, and each arrow is a state transition. The label beside each node is the logical state variable encoded in the order $x_1, x_2, \ldots, x_6$. Colors mark different basins of attraction. Attractors are highlighted using bold lines. References to color refer to the online version of this figure

which satisfies Theorem 3. Finally, one can obtain the logical dynamics of output feedback stabilizers as follows:

$$\begin{cases} \boldsymbol{x}_1(t+1) = \boldsymbol{u}_1(t) \vee [\neg \boldsymbol{x}_2(t) \wedge \neg \boldsymbol{x}_4(t)], \\ \boldsymbol{u}_1(t) = \neg \boldsymbol{y}_2(t), \\ \boldsymbol{x}_4(t+1) = \boldsymbol{u}_4(t) \wedge \boldsymbol{x}_2(t), \\ \boldsymbol{u}_4(t) = \neg \boldsymbol{y}_1(t). \end{cases} \tag{22}$$

Then under the above output feedback stabilizer (22) imposed on nodes $\boldsymbol{x}_1$ and $\boldsymbol{x}_4$, system (19) will achieve global stabilization. Fig. 4 shows the state transition digraph of system (19) under the output feedback stabilizer (22). It also shows that system (19) under the output feedback stabilizer (22) achieves global stabilization.

Remark 4 Note that during the procedures of designing an output feedback stabilizer, there is no need to calculate the state transition matrix $\boldsymbol{L}$. While using the proposed method in Li and Wang (2013), one needs to first determine matrix $\boldsymbol{L}$, which is given below

$$\begin{aligned} \boldsymbol{L} = \boldsymbol{\delta}_{64}[&52, 52, 52, 52, 52, 52, 52, 52, 52, 52, 52, 52, \\ &51, 51, 51, 51, 56, 56, 56, 56, 22, 22, 22, 22, 56, 56, \\ &56, 56, 21, 21, 21, 21, 59, 59, 59, 59, 59, 59, 59, 59, \\ &41, 41, 41, 41, 41, 41, 41, 41, 64, 64, 64, 64, 30, 30, \\ &30, 30, 46, 46, 46, 46, 13, 13, 13, 13]. \end{aligned}$$

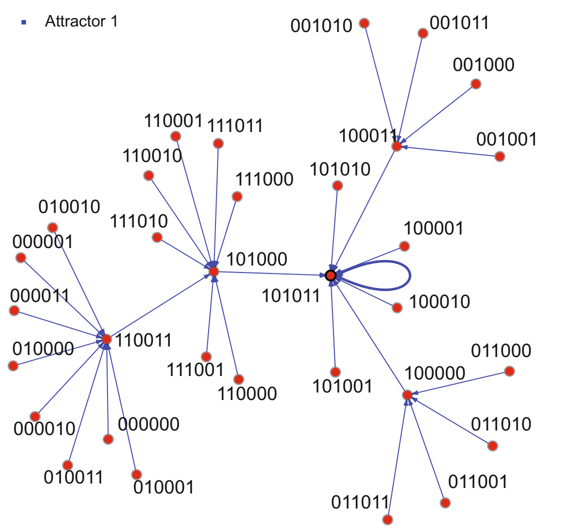

Fig. 4 State transition digraph of system (19) under the output feedback stabilizer (22), which has a unique steady state. Each node represents a state of the network, and each arrow is a state transition. The label beside each node is the logical state variable encoded in the order $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_6$. Attractor is highlighted using bold lines

Fortunately, in this study, one needs to handle only a matrix with a largest dimension of 2×16 using the proposed method. Thus, compared with the traditional control design of output feedback stabilizers (Li and Wang, 2013) using Fig. 3, the design of controllers proposed in this study is based on NS shown in Fig. 2 but not on the state transition digraph shown in Fig. 3. Thus, the computational complexity is reduced.

5 Conclusions

In this paper, an output feedback stabilizer design for global stabilization of BNs has been presented based on the NS of BNs, without using the state transition matrix. By deleting a certain number of edges such that the NS is acyclic, global stability can be guaranteed. Then based on the acyclic structure, an output feedback pinning control can be designed, where the controllers are not designed by the state transition matrix of BNs. The output feedback stabilizer design presented provides a new way to design pinning control with low dimension and calculations. Finally, we have presented a reduced signal transduction network and a reduced D. melanogaster segmentation polarity gene network to illustrate the efficiency of our main results. In this study, an efficient pinning control approach has been proposed to achieve global stabilization of BNs. In the near future, the issue of stable configuration with respect to a target equilibrium state will be studied using output feedback controls, and how to find the minimum number of control nodes and design the corresponding controller will be considered.

Compliance with ethics guidelines

Jie ZHONG, Bo-wen LI, Yang LIU, and Wei-hua GUI declare that they have no conflict of interest.

References

Aracena J, 2008. Maximum number of fixed points in regulatory Boolean networks. *Bull Math Biol*, 70(5):1398-1409. https://doi.org/10.1007/s11538-008-9304-7

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

Bang-Jensen J, Gutin G, 2008. Digraphs: Theory, Algorithms and Applications. Springer New York, USA.

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

Campbell C, Albert R, 2014. Stabilization of perturbed Boolean network attractors through compensatory interactions. *BMC Syst Biol*, 8, Article 53.
https://doi.org/10.1186/1752-0509-8-53

Cheng DZ, Liu T, 2016. A survey on logical control systems. *Unman Syst*, 4(1):97-116.
https://doi.org/10.1142/S2301385016400100

Cheng DZ, Qi HS, Li ZQ, 2010. Analysis and Control of Boolean Networks: a Semi-tensor Product Approach. Springer London, UK.

Cheng DZ, Qi HS, Li ZQ, et al., 2011. Stability and stabilization of Boolean networks. *Int J Robust Nonlin Contr*, 21(2):134-156. https://doi.org/10.1002/rnc.1581

Fan HB, Feng JE, Meng M, et al., 2018 General decomposition of fuzzy relations: semi-tensor product approach. *Fuzzy Sets Syst*, in press,
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

Guo Y, Wang P, Gui W, et al., 2015. Set stability and set stabilization of Boolean control networks based on invariant subsets. *Automatica*, 61:106-112.
https://doi.org/10.1016/j.automatica.2015.08.006

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

Kauffman S, Peterson C, Samuelsson B, et al., 2003. Random Boolean network models and the yeast transcriptional network. *PNAS*, 100(25):14796-14799.
https://doi.org/10.1073/pnas.2036429100

Kobayashi K, Hiraishi K, 2017. Design of probabilistic Boolean networks based on network structure and steady-state probabilities. *IEEE Trans Neur Netw Learn Syst*, 28(8):1966-1971.
https://doi.org/10.1109/TNNLS.2016.2572063

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

Li BW, Lu JQ, Zhong J, et al., 2019a. Fast-time stability of temporal Boolean networks. *IEEE Trans Neur Netw Learn Syst*, 30(8):2285-2294.
https://doi.org/10.1109/TNNLS.2018.2881459

Li BW, Lu JQ, Liu Y, et al., 2019b. The outputs robustness of Boolean control networks via pinning control. *IEEE Trans Contr Netw Syst*, in press.
https://doi.org/10.1109/TCNS.2019.2913543

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

Li FF, 2015. Pinning control design for the stabilization of Boolean networks. *IEEE Trans Neur Netw Learn Syst*, 27(7):1585-1590.
https://doi.org/10.1109/TNNLS.2015.2449274

Li FF, 2016. Pinning control design for the synchronization of two coupled Boolean networks. *IEEE Trans Circ Syst II*, 63(3):309-313.
https://doi.org/10.1109/TCSII.2015.2482658

Li HT, Wang YZ, 2013. Output feedback stabilization control design for Boolean control networks. *Automatica*, 49(12):3641-3645.
https://doi.org/10.1016/j.automatica.2013.09.023

Li HT, Wang YZ, 2017. Further results on feedback stabilization control design of Boolean control networks. *Automatica*, 83:303-308.
https://doi.org/10.1016/j.automatica.2017.06.043

Li R, Yang M, Chu TG, 2013. State feedback stabilization for Boolean control networks. *IEEE Trans Autom Contr*, 58(7):1853-1857.
https://doi.org/10.1109/TAC.2013.2238092

Li XD, Ho DWC, Cao JD, 2019. Finite-time stability and settling-time estimation of nonlinear impulsive systems. *Automatica*, 99:361-368.
https://doi.org/10.1016/j.automatica.2018.10.024

Li YY, Zhong J, Lu JQ, et al., 2017. On robust synchronization of drive-response Boolean control networks with disturbances. *Math Probl Eng*, 2018, Article 1737685.
https://doi.org/10.1155/2018/1737685

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

Li YY, Lou JG, Wang Z, et al., 2018b. Synchronization of dynamical networks with nonlinearly coupling function under hybrid pinning impulsive controllers. *J Franklin Inst*, 355(14):6520-6530.
https://doi.org/10.1016/j.jfranklin.2018.06.021

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, Li ML, Liu Y, et al., 2018a. Nonsingularity of Grain-like cascade FSRs via semi-tensor product. *Sci China Inform Sci*, 61:010204.
https://doi.org/10.1007/s11432-017-9269-6

Lu JQ, Sun LJ, Liu Y, et al., 2018b. 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., 2018c. 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

Mao Y, Wang L, Liu Y, et al., 2018. Stabilization of evolutionary networked games with length-r information. *Appl Math Comput*, 337:442-451.
https://doi.org/10.1016/j.amc.2018.05.027

Meng M, Lam J, Feng J, et al., 2018. Stability and guaranteed cost analysis of time-triggered Boolean networks. *IEEE Trans Neur Netw Learn Syst*, 29(8):3893-3899.
https://doi.org/10.1109/TNNLS.2017.2737649

Mori F, Mochizuki A, 2017. Expected number of fixed points in Boolean networks with arbitrary topology. *Phys Rev Lett*, 119:028301.
https://doi.org/10.1103/PhysRevLett.119.028301

Murrugarra D, Veliz-Cuba A, Aguilar B, et al., 2016. Identification of control targets in Boolean molecular network models via computational algebra. *BMC Syst Biol*, 10, Article 94.
https://doi.org/10.1186/s12918-016-0332-x

Pan J, Feng J, Meng M, 2019. Steady-state analysis of probabilistic Boolean networks. *J Franklin Inst*, 356(5):2994-3009. https://doi.org/10.1016/j.jfranklin.2019.01.039

Paulevé L, Richard A, 2012. Static analysis of Boolean networks based on interaction graphs: a survey. *Electron Notes Theor Comput Sci*, 284:93-104.
https://doi.org/10.1016/j.entcs.2012.05.017

Robert F, 1986. Discrete Iterations: a Metric Study. Springer New York, USA.

Saadatpour A, Albert I, Albert R, 2010. Attractor analysis of asynchronous Boolean models of signal transduction networks. *J Theor Biol*, 266(4):641-656.
https://doi.org/10.1016/j.jtbi.2010.07.022

Saadatpour A, Wang R, Liao A, et al., 2011. Dynamical and structural analysis of a T cell survival network identifies novel candidate therapeutic targets for large granular lymphocyte leukemia. *PLoS Comput Biol*, 7(11):e1002267.
https://doi.org/10.1371/journal.pcbi.1002267

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

Wang B, Feng J, 2019. On detectability of probabilistic Boolean networks. *Inform Sci*, 483:383-395.
https://doi.org/10.1016/j.ins.2019.01.055

Wu Y, Shen T, 2018. Policy iteration algorithm for optimal control of stochastic logical dynamical systems. *IEEE Trans Neur Netw Learn Syst*, 29(5):2031-2036.
https://doi.org/10.1109/TNNLS.2017.2661863

Xiao YF, Dougherty ER, 2007. The impact of function perturbations in Boolean networks. *Bioinformatics*, 23(10):1265-1273.
https://doi.org/10.1093/bioinformatics/btm093

Yang M, Li R, Chu TG, 2013. Controller design for disturbance decoupling of Boolean control networks. *Automatica*, 49(1):273-277.
https://doi.org/10.1016/j.automatica.2012.10.010

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

Zhang K, Zhang L, 2016. Observability of Boolean control networks: a unified approach based on finite automata. *IEEE Trans Autom Contr*, 61(9):2733-2738.
https://doi.org/10.1109/TAC.2015.2501365

Zhao Y, Ghosh BK, Cheng D, 2016. Control of large-scale Boolean networks via network aggregation. *IEEE Trans Neur Netw Learn Syst*, 27(7):1527-1536.
https://doi.org/10.1109/TNNLS.2015.2442593

Zhu QX, Lin W, 2019. Stabilizing Boolean networks by optimal event-triggered feedback control. *Syst Contr Lett*, 126:40-47.
https://doi.org/10.1016/j.sysconle.2019.03.002

Zhu QX, Liu Y, Lu J, et al., 2018. On the optimal control of Boolean control networks. *SIAM J Contr Optim*, 56(2):1321-1341. https://doi.org/10.1137/16M1070281

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, Article 9259348.
https://doi.org/10.1155/2018/9259348