

# A polynomial-time criterion for the stability of discrete iteration systems under deterministic asynchronous update

Wenrong LI &amp; Haitao LI\*

*School of Mathematics and Statistics, Shandong Normal University, Jinan 250014, China*

Received 2 December 2024/Revised 28 April 2025/Accepted 13 June 2025/Published online 4 June 2026

**Abstract** Discrete iteration systems (DISs) are fundamental for verifying the effectiveness of iterative algorithms in the simulation of nonlinear systems typically used in gene regulatory networks and finite-field networks. In this study, we explore the stability of DISs with deterministic asynchronous update (DAU) from a graph-theoretic perspective. First, a polynomial-time criterion for the finite-time stability of DISs subject to DAU is constructed using layer sets in a network graph. Moreover, an algorithm is constructed for seeking out the equilibria. Second, an edge removal control is designed to stabilize DISs with DAU based on a minimal feedback arc set of the network graph, and the desired equilibrium can be reached by adjusting some parameters in the mixed state domains. Finally, our results are demonstrated using a finite-field network and a pancreatic cancer network with 63 nodes as examples.

**Keywords** discrete iteration systems, deterministic asynchronous update, finite-time stability, edge removal control, feedback arc set

**Citation** Li W R, Li H T. A polynomial-time criterion for the stability of discrete iteration systems under deterministic asynchronous update. *Sci China Inf Sci*, 2026, 69(7): 172203, <https://doi.org/10.1007/s11432-024-4869-5>

## 1 Introduction

Owing to the high efficiency of iterative algorithms in the simulation for nonlinear systems, discrete iteration systems (DISs) have been developed in diverse fields, such as gene regulatory networks [1], game theory [2], finite-field networks [3], finite state machines [4], feedback shift registers [5], and control engineering [6]. A typical characteristic of DISs is that all variables are assigned in a family of finite sets [7]. In this case, one can explore DISs using Boolean networks (BNs), multi-valued logical networks, and mix-valued logical networks (MLNs) [8]. In the past few decades, numerous approaches have been proposed to explore the dynamical behavior of DISs, such as metric tool [7], polynomial algebra [9], semi-tensor product [10,11], and network structure approaches [12,13].

Most existing research outcomes of DISs center around synchronous updates [14,15]; i.e., all nodes update synchronously according to their inherent dynamical characteristics. The need to describe different time scales motivates the development of various asynchronous update rules, including random asynchronous update, general asynchronous update, and deterministic asynchronous update (DAU) [16]. Particularly, Ref. [17] showed that DAU can effectively describe the basic structure and function of gene regulatory networks. Owing to the effect of update rules on node dynamics, the dynamical behavior and control design of systems with asynchronous updates differ from those of synchronous update schemes [18].

Several fundamental results have been proposed to analyze the attractors, stability, and fault detectability of asynchronous DISs. With the help of negative circuits in the interaction graph, the oscillatory behaviors of asynchronous MLNs were investigated in [19]. Based on the positive and negative cycles of network graphs, the attractors of asynchronous BNs were discussed in [20]. A target control has been proposed for the global stability of asynchronous BNs when the network graph can be decomposed into several blocks [21]. Using the semi-tensor product of matrices, the state feedback stability and fault detectability of asynchronous BNs were solved in [22,23], respectively. The computational complexity of the reachability problem of asynchronous MLNs was proven to be NP-complete in [24]. By using an SAT-based attractor finder, the attractors of asynchronous MLNs were computed in [25], whose computational complexity is exponential.

For synchronous DISs, Robert [7] showed that systems with an acyclic network graph can achieve finite-time stability. However, Robert [7] highlighted that the metric tool is not directly applicable to asynchronous DISs

\* Corresponding author (email: [lihaitao\\_sdu@163.com](mailto:lihaitao_sdu@163.com))

and cannot construct the corresponding equilibrium, which is thus left for future work. Combining the findings of Robert [7] and the semi-tensor product of matrices, the pinning control method [26] was introduced to achieve the stability of asynchronous BNs based on a network graph. However, in the worst case, the computational complexity of pinning control is exponential in the size of MLNs.

Some disease-causing mutations in humans can be modeled as node and edge removal of gene regulatory networks and may pave the way for novel molecular therapeutic strategies [27]. The edge removal can be induced in vivo via small molecule drug inhibitors [28] or CRISPR-based gene editing [29]. In [28], cancerous attractors were eliminated via the edge perturbations of synchronous BNs. Based on the Boolean operators, an edge removal control was introduced in [30] to delete edges and achieve the asymptotical stability of asynchronous BNs. Notably, the edge removal technique proposed in [30] does not apply to logical operators among mixed state domains because the logical operations between BNs and MLNs are quite different; thus, it is not appropriate for asynchronous MLNs.

In this study, we discuss the stability of DISs subject to DAU by using the graph-theoretic framework. The major contributions of this study are as follows.

(i) We analyze the finite-time stability and precisely determine the equilibrium of DISs with DAU by constructing the layer sets for an acyclic network graph. Compared with the metric tool in [7], our method is not only suitable for asynchronous updating but can also construct stable equilibria.

(ii) Using the feedback arc set of network graphs, we design an edge removal control for the stabilization of DISs with DAU. The computational complexity of designing the edge removal control is  $O(n^3)$ , which overcomes the exponential complexity of the pinning control method [26].

(iii) By adjusting proper parameters and equipping suitable operators in the mixed state domains, we generalize the edge removal control from BNs [30] to MLNs. The new edge removal control technique provides a theoretical framework for studying MLN control-related problems.

The remainder of this article is organized as follows. Section 2 provides some preliminary results and formulates the problem. In Section 3, we present a polynomial-time criterion for the finite-time stability of DISs with DAU and discuss the edge removal control design to achieve stability. Section 4 applies the main results to a finite-field network and a pancreatic cancer network. Section 5 presents a brief conclusion of this study.

## 2 Preliminaries and problem formulation

### 2.1 Preliminaries

Some useful notations used throughout this article are listed below.

Let  $(A)_{i,j}$ ,  $Row_i(A)$ , and  $Col_j(A)$  denote the  $(i, j)$ -th element,  $i$ -th row, and  $j$ -th column of matrix  $A$ , respectively.  $\mathcal{B}_{n \times m}$  denotes the set of  $n \times m$  Boolean matrices.  $[i, j]_{\mathbb{N}} := \{i, i+1, \dots, j\}$ , where  $i, j \in \mathbb{N}$  and  $i < j$ .  $\mathcal{D}_k := [0, k-1]_{\mathbb{N}}$ ,  $k \geq 2$ . Given  $a, b \in \mathcal{D}_k$ , let  $a \vee_k b = \max\{a, b\}$  and  $a \wedge_k b = \min\{a, b\}$ .  $\mathbf{1}_n = [\underbrace{1 \dots 1}_n]^\top$ .  $\mathbf{0}_n = [\underbrace{0 \dots 0}_n]^\top$ .  $\nabla_{i,k}$

is a unary operator on  $\mathcal{D}_k$ , which satisfies

$$\nabla_{i,k}(a) = \begin{cases} k-1, & \text{if } a = k-i, \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

As in [7], the DIS dynamics is given by

$$\begin{cases} X_1(t+1) = f_1(X_1(t), \dots, X_n(t)), \\ \vdots \\ X_n(t+1) = f_n(X_1(t), \dots, X_n(t)), \end{cases} \quad (2)$$

where  $X_i(t) \in \mathcal{D}_{k_i}$  is the state of node  $i$  at time  $t$ , and  $f_i : \prod_{j=1}^n \mathcal{D}_{k_j} \rightarrow \mathcal{D}_{k_i}$  is the update function. The state vector of system (2) at time  $t$  is represented by a vector  $X(t) = (X_1(t), \dots, X_n(t))^\top$ . Then, system (2) can be rewritten as follows:

$$X(t+1) = F(X(t)), \quad (3)$$

where  $F : \prod_{i=1}^n \mathcal{D}_{k_i} \rightarrow \prod_{i=1}^n \mathcal{D}_{k_i}$  is a mix-valued logical mapping. Notably, all nodes in system (2) are updated synchronously.

The dependency relationship among the state nodes in system (2) can be described by the incidence matrix  $\mathcal{I} \in \mathcal{B}_{n \times n}$ , where

$$(\mathcal{I})_{i,j} = \begin{cases} 1, & \text{if } f_i \text{ depends on } X_j, \\ 0, & \text{otherwise.} \end{cases} \quad (4)$$

The incidence matrix  $\mathcal{I}$  uniquely determines the network graph of system (2), denoted by  $\mathcal{W} = (\mathcal{N}, \mathcal{E})$ , where  $\mathcal{N} = [1, n]_{\mathbb{N}}$  and  $\mathcal{E} = \{j \rightarrow i : (\mathcal{I})_{i,j} = 1\}$ . If  $(\mathcal{I})_{i,j} = 1$ , then node  $j$  is said to be the parent node of node  $i$ , and node  $i$  is said to be the child node of node  $j$ . Let  $\mathcal{P}_i$  and  $\mathcal{C}_i$  denote the sets of parent and child nodes for node  $i \in \mathcal{N}$ , respectively.

## 2.2 Problem formulation

DAU has been widely applied in the study of BNs [22], sequential machines [31], and automata [32]. In this study, we are interested in DISs subject to DAU, where each node's update is associated with a preselected update period and an initial updating time.

Assume that the updating period and initial updating time of node  $i$  are  $\rho_i \in \mathbb{Z}_+$  and  $\eta_i \in [0, \rho_i - 1]_{\mathbb{N}}$ , respectively,  $i \in \mathcal{N}$ . The dynamics of a DIS with DAU is as follows:

$$X_i(t+1) = \begin{cases} f_i(X_1(t), \dots, X_n(t)), & \text{if } \varphi_i(t+1) = 1, \\ X_i(t), & \text{if } \varphi_i(t+1) = 0, \end{cases} \quad (5)$$

where

$$\varphi_i(t) = \begin{cases} 1, & \text{if } t \bmod \rho_i = \eta_i, \\ 0, & \text{otherwise.} \end{cases} \quad (6)$$

For node  $i$ , if  $\varphi_i(t+1) = 1$ , then the value of  $X_i(t+1)$  is determined by the update function  $f_i$ ; otherwise, the value of  $X_i(t+1)$  remains unchanged. The network graph for system (5) is the same as that for system (2).

For BNs with DAU, a trap domain is a significant concept [33]. Naturally, we extend this concept to DISs subject to DAU. A nonempty set  $\mathcal{T}_d \subseteq \prod_{i=1}^n \mathcal{D}_{k_i}$  is said to be a trap domain of system (5) if  $X(1; X(t)) \in \mathcal{T}_d$  holds for any  $X(t) \in \mathcal{T}_d$  and  $t \in \mathbb{N}$ . A trap domain  $\mathcal{A} \subseteq \prod_{i=1}^n \mathcal{D}_{k_i}$  is called an attractor if any nonempty proper subset of  $\mathcal{A}$  is not a trap domain. An attractor  $\mathcal{A}$  with  $|\mathcal{A}| = 1$  is said to be a fixed point; otherwise,  $\mathcal{A}$  is called a cyclic attractor.

As highlighted in [22], if system (5) has a fixed point as the unique attractor, then it is finite-time stable. We present another formal definition of finite-time stability for system (5).

**Definition 1.** System (5) is said to be finite-time stable at  $X_e = (X_1^e, \dots, X_n^e) \in \prod_{i=1}^n \mathcal{D}_{k_i}$ , if for any initial state  $X_0 \in \prod_{i=1}^n \mathcal{D}_{k_i}$ , there exists a positive integer  $T$  such that

$$X(t; X_0) = X_e$$

holds for any integer  $t \geq T$ .

By proposing the metric tool, Robert [7] demonstrated that synchronous DISs are finite-time stable when the network graphs are acyclic. Nevertheless, the metric tool cannot be directly generalized to asynchronous iterations. In the sequel, we will develop a new tool to show that the finding in [7] applies to DISs under DAU.

## 3 Main results

In this section, we aim to prove that a DIS under DAU is finite-time stable when its network graph is acyclic and precisely seek out the fixed point. Then, we use the edge removal control to study the stability of system (5) based on the feedback arc set of the network graph  $\mathcal{W} = (\mathcal{N}, \mathcal{E})$ .

### 3.1 Finite-time stability analysis

For an acyclic network graph, we can assign a unique layer to each node  $i \in \mathcal{N}$  in light of several layer sets.

With the help of the incidence matrix  $\mathcal{I}$ , we define the layer sets

$$\mathfrak{L}_0 = \left\{ i \in \mathcal{N} : \text{Row}_i(\mathcal{I}) = \mathbf{0}_n^\top \right\}, \quad (7)$$

and

$$\mathfrak{L}_j = \left\{ i \in \mathcal{N} : \text{Row}_i(\mathcal{I})\mathcal{R}_{j-1} > 0 \text{ and } \text{Row}_i(\mathcal{I}) \left[ \mathbf{1}_n - \sum_{k=0}^{j-1} \mathcal{R}_k \right] = 0 \right\}, \quad j \in \mathbb{Z}_+, \quad (8)$$

where  $\mathcal{R}_k \in \mathcal{B}_{n \times 1}$ ,  $k \in \mathbb{N}$  and

$$(\mathcal{R}_k)_{i,1} = \begin{cases} 1, & \text{if } i \in \mathfrak{L}_k, \\ 0, & \text{otherwise.} \end{cases}$$

$\mathfrak{L}_k$  denotes the set of nodes which belong to the  $k$ -th layer,  $k \in \mathbb{N}$ . If node  $i \in \mathfrak{L}_j$ , then there must be a parent node belonging to  $\mathfrak{L}_{j-1}$  and all the parent nodes of node  $i$  are part of  $\bigcup_{k=0}^{j-1} \mathfrak{L}_k$ ,  $j \in \mathbb{Z}_+$ , that is,

$$\mathcal{P}_i \cap \mathfrak{L}_{j-1} \neq \emptyset, \text{ and } \mathcal{P}_i \subseteq \bigcup_{k=0}^{j-1} \mathfrak{L}_k, \quad (9)$$

where  $\mathcal{P}_i$  is the set of parent nodes for node  $i$ . Because the network graph is acyclic, it is easy to obtain that  $\mathfrak{L}_{k_1} \cap \mathfrak{L}_{k_2} = \emptyset$ ,  $k_1 \neq k_2$ ,  $k_1, k_2 \in \mathbb{N}$ .

**Proposition 1.** Assume that there exists an integer  $j^* \in [0, n-1]_{\mathbb{N}}$  satisfying  $\mathfrak{L}_{j^*} \neq \emptyset$  and  $\mathfrak{L}_{j^*+1} = \emptyset$ . Then, it holds that

$$\text{Col}_i(\mathcal{I}) = \mathbf{0}_n, \quad i \in \mathfrak{L}_{j^*}. \quad (10)$$

*Proof.* We first prove  $\mathfrak{L}_j = \emptyset$ ,  $\forall j > j^*$  by induction. Apparently,  $\mathfrak{L}_{j^*+1} = \emptyset$ . Assume that  $\mathfrak{L}_{j^*+k} = \emptyset$ ,  $k \in \mathbb{Z}_+$ . According to (8), one has  $\text{Row}_i(\mathcal{I})\mathcal{R}_{j^*+k} = 0$ , which means that  $\mathfrak{L}_{j^*+k+1} = \emptyset$ . Hence,  $\mathfrak{L}_j = \emptyset$ ,  $\forall j > j^*$ . Consequently,

$$\bigcup_{k=0}^{j^*} \mathfrak{L}_k = \mathcal{N}, \quad j^* \in [0, n-1]_{\mathbb{N}}.$$

Noting that

$$\mathfrak{L}_{k_1} \cap \mathfrak{L}_{k_2} = \emptyset, \quad k_1 \neq k_2, \quad k_1, k_2 \in [0, j^*]_{\mathbb{N}},$$

one has  $\mathcal{R}_0 \circ \cdots \circ \mathcal{R}_{j^*} = \mathbf{0}_n$  and  $\sum_{k=0}^{j^*} \mathcal{R}_k = \mathbf{1}_n$ , where  $\circ$  is the Hadamard product of matrices.

For any node  $i \in \mathfrak{L}_{j^*}$ , we assume that there exists an integer  $a \in \mathfrak{L}_j$ ,  $j \in [0, j^*]_{\mathbb{N}}$  such that  $(\mathcal{I})_{a,i} = 1$ . One derives from  $\mathcal{I} \in \mathcal{B}_{n \times n}$  and (8) that

$$\begin{aligned} 0 &= \text{Row}_a(\mathcal{I}) \left[ \mathbf{1}_n - \sum_{k=0}^{j-1} \mathcal{R}_k \right] \geq \text{Row}_a(\mathcal{I}) \left[ \mathbf{1}_n - \sum_{k=0}^{j^*-1} \mathcal{R}_k \right] \\ &= \text{Row}_a(\mathcal{I})\mathcal{R}_{j^*} = \sum_{l=1}^n (\mathcal{I})_{a,l} (\mathcal{R}_{j^*})_{l,1} \geq (\mathcal{I})_{a,i} = 1, \end{aligned}$$

which is a contradiction. Therefore,  $\text{Col}_i(\mathcal{I}) = \mathbf{0}_n$ .

Based on Proposition 1, if  $\mathfrak{L}_{j^*} \neq \emptyset$  and  $\mathfrak{L}_{j^*+1} = \emptyset$ , then

$$\mathfrak{L}_{j^*} = \left\{ i \in \mathcal{N} : \text{Row}_i(\mathcal{I})\mathcal{R}_{j-1} > 0 \text{ and } \text{Col}_i(\mathcal{I}) = \mathbf{0}_n \right\}.$$

According to the layer sets (7) and (8), we analyze the finite-time stability of DISs under DAU.

**Theorem 1.** System (5) is finite-time stable if its network graph  $\mathcal{W} = (\mathcal{N}, \mathcal{E})$  is acyclic.

*Proof.* Given an arbitrary node  $i \in \mathfrak{L}_k$ ,  $k \in [0, j^*]_{\mathbb{N}}$ , we prove by induction that there exists a constant value  $\xi_i \in \mathcal{D}_{k_i}$  such that

$$X_i(t; X(0)) = \xi_i \quad (11)$$

holds for any  $X(0) \in \prod_{i=1}^n \mathcal{D}_{k_i}$  and any integer  $t \geq \max_{j \in \mathcal{N}} \{\eta_j + k\rho_j\}$ , where  $\eta_j$  and  $\rho_j$  are given in (5).

When  $k = 0$ , one has  $\text{Row}_i(\mathcal{I}) = \mathbf{0}_n^\top$ , which together with (4) implies that the update function  $f_i$  is constant. Therefore, there exists a constant value  $\xi_i \in \mathcal{D}_{k_i}$  such that

$$f_i(X) := \xi_i, \quad \forall X \in \prod_{i=1}^n \mathcal{D}_{k_i}.$$

One derives from the asynchronous update rule that

$$X_i(t; X(0)) = \begin{cases} X_i(0), & \text{if } t < \eta_i, \\ f_i(X(t-1; X(0))) = \xi_i, & \text{if } t \geq \eta_i. \end{cases}$$

As a result,

$$X_i(t; X(0)) = \xi_i, \quad \forall X(0) \in \prod_{i=1}^n \mathcal{D}_{k_i}, \quad \forall t \geq \max_{j \in \mathcal{N}} \{\eta_j\}.$$

Assume that Eq. (11) holds for any integer  $k \in [0, s]_{\mathbb{N}}$ . Let

$$j' = \arg \max_{j \in \mathcal{N}} \{\eta_j + s\rho_j\}.$$

Now, we consider the case of  $k = s + 1 \leq j^*$ . When  $i \in \mathfrak{L}_{s+1}$ , there exists a smallest positive integer  $T$  satisfying  $T > j'$  and  $\varphi_i(T) = 1$ , where  $\varphi_i$  is defined in (6). Hence, there exists an integer  $m \geq s + 1$  such that  $T = \eta_i + m\rho_i$ . Considering the minimality of  $T$ , if  $m \geq s + 2$ , then

$$(m-1)\rho_i \leq \eta_i + (m-1)\rho_i \leq \eta_{j'} + s\rho_{j'} < (s+1)\rho_{j'}.$$

Because  $m-1 \geq s+1$ , one obtains that  $\rho_i < \rho_{j'}$  and

$$T = \eta_i + (m-1)\rho_i + \rho_i \leq \eta_{j'} + s\rho_{j'} + \rho_{j'} = \eta_{j'} + (s+1)\rho_{j'} \leq \max_{j \in \mathcal{N}} \{\eta_j + (s+1)\rho_j\}.$$

If  $m = s + 1$ , it is obvious that

$$T = \eta_i + (s+1)\rho_i \leq \max_{j \in \mathcal{N}} \{\eta_j + (s+1)\rho_j\}.$$

Hence,

$$T \leq \max_{j \in \mathcal{N}} \{\eta_j + (s+1)\rho_j\}. \quad (12)$$

According to (8), one has  $\mathcal{P}_i \cap \mathfrak{L}_s \neq \emptyset$  and  $\mathcal{P}_i \subseteq \bigcup_{k=0}^s \mathfrak{L}_k$ . If  $t \geq T$  and  $\varphi_i(t) = 1$ , then it holds that

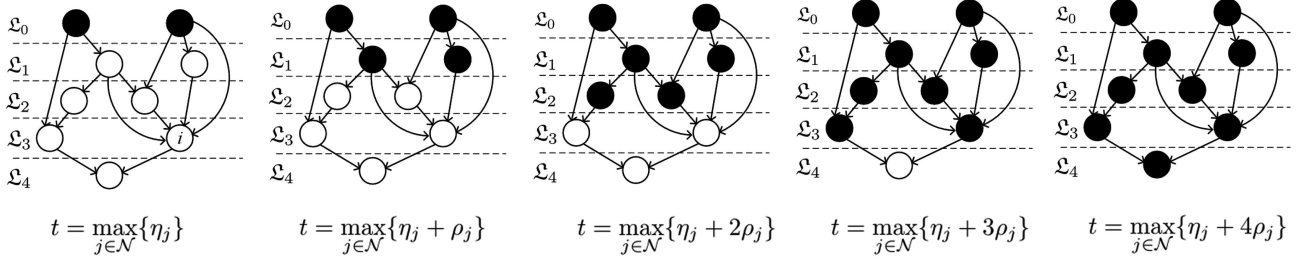
$$X_i(t; X(0)) = f_i\left([X_j(t-1; X(0))]_{j \in \mathcal{P}_i}\right) = f_i\left([\xi_j]_{j \in \mathcal{P}_i}\right) := \xi_i, \quad \forall X(0) \in \prod_{i=1}^n \mathcal{D}_{k_i}. \quad (13)$$

When  $t > T$  and  $\varphi_i(t) = 0$ , there exists a positive integer  $t'$  satisfying  $0 < t - t' < \rho_i$  and  $\varphi_i(t') = 1$  such that

$$X_i(t; X(0)) = X_i(t'; X(0)).$$

Because  $T$  is the smallest positive integer satisfying  $T > j'$  and  $\varphi_i(T) = 1$ , it holds that  $t' \geq T$ . From (13), we obtain

$$X_i(t; X(0)) = X_i(t'; X(0)) := \xi_i, \quad \forall X(0) \in \prod_{i=1}^n \mathcal{D}_{k_i}. \quad (14)$$



**Figure 1** State evolution for nodes in a DIS with an acyclic network graph, where the solid dot represents that the state of the node must be fixed.

Therefore, combining (12)–(14), we have

$$X_i(t; X(0)) = \xi_i, \forall X(0) \in \prod_{i=1}^n \mathcal{D}_{k_i}, \forall t \geq \max_{j \in \mathcal{N}} \{\eta_j + (s+1)\rho_j\}.$$

In summary, let

$$T^* = \max_{j \in \mathcal{N}} \{\eta_j + j^* \rho_j\} \leq n \max_{j \in \mathcal{N}} \rho_j.$$

For any initial state  $X(0) \in \prod_{i=1}^n \mathcal{D}_{k_i}$ , we have

$$X(t; X(0)) = (\xi_1, \dots, \xi_n), \forall t \geq T^*.$$

Thus, system (5) is finite-time stable.

**Remark 1.** When the network graph  $\mathcal{W} = (\mathcal{N}, \mathcal{E})$  is acyclic, one can construct the fixed point  $X_e$  of system (5) from the proof of Theorem 1. Then, Algorithm 1 is established to find  $X_e$ . Although the finite-time stability of synchronous DISs was well addressed by Robert [7], the corresponding equilibrium was not constructed. Actually, Algorithm 1 also applies to the synchronous DISs, which can be used to precisely seek out equilibria.

**Remark 2.** The set of parent nodes for node  $i \in \mathcal{L}_j$ ,  $\mathcal{P}_i \subseteq \bigcup_{k=0}^{j-1} \mathcal{L}_k$  may come from different layers. Specifically, if a node  $i$  in layer  $\mathcal{L}_j$  has parent nodes in both  $\mathcal{L}_{j-1}$  and an earlier layer, then the states of its parent nodes will be determined layer by layer. Afterward, the state of node  $i$  is determined according to the states of its parent nodes. Consider node  $i \in \mathcal{L}_3$  as an example, which has parent nodes in  $\mathcal{L}_0$ ,  $\mathcal{L}_1$  and  $\mathcal{L}_2$  (see Figure 1). When  $t = \max_{j \in \mathcal{N}} \{\eta_j + 2\rho_j\}$ , we know from Theorem 1 that  $X_i(t; X(0)) = \xi_i, \forall X(0) \in \prod_{i=1}^n \mathcal{D}_{k_i}, \forall i \in \mathcal{L}_0 \cup \mathcal{L}_1 \cup \mathcal{L}_2$ . If the positive integer  $T$  satisfies  $\max_{j \in \mathcal{N}} \{\eta_j + 2\rho_j\} < T \leq \max_{j \in \mathcal{N}} \{\eta_j + 3\rho_j\}$  and  $\varphi_i(T) = 1$ , then

$$X_i(T) = f_i([X_k(T-1)]_{k \in \mathcal{L}_0 \cup \mathcal{L}_1 \cup \mathcal{L}_2}) = f_i([\xi_k]_{k \in \mathcal{L}_0 \cup \mathcal{L}_1 \cup \mathcal{L}_2}) = \xi_i.$$

Hence, the state of node  $i$  at time  $T$  is determined by its parent nodes in different layers  $\mathcal{L}_0$ ,  $\mathcal{L}_1$  and  $\mathcal{L}_2$ .

**Remark 3.** If  $X_i(t) \in \mathcal{D}_2$ , then system (5) degenerates into deterministic asynchronous BNs. In [22], the algebraic state space method provides a general criterion for the stability of BNs under DAU. For a deterministic asynchronous BN with an acyclic network graph, the upper bound guaranteeing the finite-time stability is  $2^n$  via the criterion in [22]. Meanwhile, from the proof of Theorem 1, the network graph can be divided into at most  $n-1$  layers, and thus such an upper bound becomes  $\max_{i \in \mathcal{N}} \{\eta_i + (n-1)\rho_i\} \leq \max_{i \in \mathcal{N}} \{\rho_i + (n-1)\rho_i\} = n \max_{i \in \mathcal{N}} \{\rho_i\}$ , which is much less than  $2^n$  because  $n \max_{i \in \mathcal{N}} \{\rho_i\} = O(n)$ .

**Example 1.** Consider the following DIS:

$$\begin{cases} X_1(t+1) = X_2(t) \vee_3 X_4(t), \\ X_2(t+1) = \nabla_{2,4}(X_4(t)), \\ X_3(t+1) = X_1(t) \wedge_3 X_4(t) \wedge_3 X_5(t), \\ X_4(t+1) = 2, \\ X_5(t+1) = X_1(t) \vee_3 X_2(t), \\ X_6(t+1) = X_1(t) \wedge_3 X_5(t), \\ X_7(t+1) = \nabla_{2,3}(X_1(t)), \end{cases} \quad (15)$$

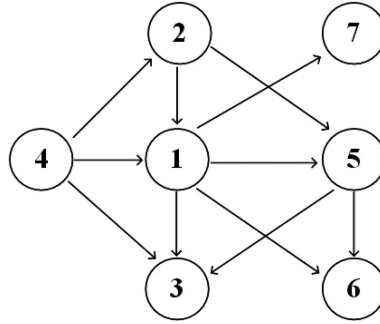
---

**Algorithm 1** Computing the fixed point of system (5) in an acyclic network graph.
 

---

**Require:**  $\mathcal{I}, f_i, i \in \mathcal{N}, \mathcal{L}_j \Leftarrow \emptyset, \mathcal{R}_j \Leftarrow \mathbf{0}_n, j \in \{0, \dots, n\}$ ;  
**Ensure:**  $X_e = (\xi_1, \dots, \xi_n)$ ;  
 1: **for**  $i = 1, \dots, n$  **do**  
 2:   **if**  $\text{Row}_i(\mathcal{I}) = \mathbf{0}_n^\top$  **then**  
 3:      $\mathcal{L}_0 \Leftarrow \mathcal{L}_0 \cup \{i\}, (\mathcal{R}_0)_{i,1} \Leftarrow 1$ ;  
 4:   **end if**  
 5: **end for**  
 6: **for**  $j = 1, \dots, n$  **do**  
 7:   **for**  $i = 1, \dots, n$  **do**  
 8:     **if**  $\text{Row}_i(\mathcal{I})\mathcal{R}_{j-1} > 0$  **and**  $\text{Row}_i(\mathcal{I})\left[\mathbf{1}_n - \sum_{k=0}^{j-1} \mathcal{R}_k\right] = 0$  **then**  
 9:        $\mathcal{L}_j \Leftarrow \mathcal{L}_j \cup \{i\}, (\mathcal{R}_j)_{i,1} \Leftarrow 1$ ;  
 10:     **end if**  
 11:   **end for**  
 12:   **if**  $\mathcal{L}_j = \emptyset$  **then**  
 13:      $j^* \Leftarrow j - 1$ ;  
 14:     **break**  
 15:   **end if**  
 16: **end for**  
 17: **for**  $i \in \mathcal{L}_0$  **do**  
 18:    $\xi_i \Leftarrow f_i$ ;  
 19: **end for**  
 20: **for**  $i = 1, \dots, j^*$  **do**  
 21:   **for**  $j \in \mathcal{L}_i$  **do**  
 22:      $\xi_j \Leftarrow f_j\left([\xi_k]_{k \in \cup_{s=0}^{i-1} \mathcal{L}_s}\right)$ ;  
 23:   **end for**  
 24: **end for**

---



**Figure 2** Network graph of system (15).

where  $X(t) = (X_1(t), X_2(t), X_3(t), X_4(t), X_5(t), X_6(t), X_7(t)) \in \mathcal{D}_3 \times \mathcal{D}_4 \times \mathcal{D}_3 \times \mathcal{D}_4 \times \mathcal{D}_3 \times \mathcal{D}_3 \times \mathcal{D}_3$ .

The definitions of logical operators for mix-valued logical variables are of less logical meaning and messy [8]. Thus, one can take advantage of the  $q$ -valued up-round projection  $\phi_{[q,p]}$  to project the state variable in  $\mathcal{D}_p$  to  $\mathcal{D}_q$ . For example,  $\phi_{[3,4]}(x) \in \mathcal{D}_3$  satisfies  $\phi_{[3,4]}(0) = 0$ ,  $\phi_{[3,4]}(1) = 1$ ,  $\phi_{[3,4]}(2) = 1$ , and  $\phi_{[3,4]}(3) = 2$ . Then, it holds that

$$\begin{cases} X_2 \vee_3 X_4 = \phi_{[3,4]}(X_2) \vee_3 \phi_{[3,4]}(X_4), \\ X_1 \wedge_3 X_4 \wedge_3 X_5 = X_1 \wedge_3 \phi_{[3,4]}(X_4) \wedge_3 X_5, \\ X_1 \vee_3 X_2 = X_1 \vee_3 \phi_{[3,4]}(X_2). \end{cases} \quad (16)$$

Figure 2 shows the network graph of system (15). Assume that system (15) updates subject to DAU (5), where  $\eta_i = i$  and  $\rho_i = i + 2, i \in [1, 7]_{\mathbb{N}}$ .

Because the network graph of system (15) is acyclic, we assign a unique layer to each node  $i \in [1, 7]_{\mathbb{N}}$ . According to Algorithm 1, one has

$$\mathcal{L}_0 = \{4\}, \mathcal{L}_1 = \{2\}, \mathcal{L}_2 = \{1\}, \mathcal{L}_3 = \{5, 7\}, \mathcal{L}_4 = \{3, 6\}.$$

Consequently, for any initial state  $X(0)$ , it holds that

$$\begin{cases} X_4(t; X(0)) = 2, t \geq 4, \\ X_2(t; X(0)) = \nabla_{2,4}(2) = 3, t \geq 6, \\ X_1(t; X(0)) = \phi_{[3,4]}(2) \vee_3 \phi_{[3,4]}(3) = 1 \vee_2 2 = 2, t \geq 7, \\ X_5(t; X(0)) = 2 \vee_3 \phi_{[3,4]}(3) = 2 \vee_3 2 = 2, t \geq 12, \\ X_3(t; X(0)) = 2 \wedge_3 \phi_{[3,4]}(2) \wedge_3 2 = 2 \wedge_3 1 \wedge_3 2 = 1, t \geq 13, \\ X_6(t; X(0)) = 2 \wedge_3 2 = 2, t \geq 14, \\ X_7(t; X(0)) = \nabla_{2,3}(2) = 0, t \geq 16. \end{cases} \quad (17)$$

Taking  $T^* = 16$ , we conclude

$$X(t; X(0)) = (2, 3, 1, 3, 2, 2, 0), \forall t \geq T^*.$$

Thus, system (15) subject to DAU is finite-time stable at  $(2, 3, 1, 3, 2, 2, 0)$ .

### 3.2 Design of edge removal control

Assume that  $\Theta$  is the family of attractors for system (5) and  $\{X_e\} := \{(\xi_1, \dots, \xi_n)\} \in \Theta$  is a desired fixed point. In this subsection, we use the edge removal control to stabilize system (5) at the desired fixed point  $X_e$  in a finite-time manner. From Theorem 1, if we delete several edges in the network graph  $\mathcal{W} = (\mathcal{N}, \mathcal{E})$  of system (5) such that  $\mathcal{W} = (\mathcal{N}, \mathcal{E})$  is acyclic, then system (5) will be finite-time stable. First, we define the edge removal for system (5).

**Definition 2.** Edge  $j \rightarrow i \in \mathcal{E}$  in the network graph  $\mathcal{W} = \{\mathcal{N}, \mathcal{E}\}$  of system (5) is said to be removed, if  $f_i$  is independent of  $X_j$ .

In the Boolean setting, we introduce a control variable  $U_{j,i} \in \mathcal{D}_2$  for deleting an edge  $j \rightarrow i$ . Inspired by the idea in [30], for the mix-valued case, the silencing of the interaction can also be achieved by a control variable  $U_{j,i} \in \mathcal{D}_{k_j}$ .

**Proposition 2.** Consider system (5). Edge  $j \rightarrow i \in \mathcal{E}$  is removed permanently if and only if edge removal control  $U_{j,i} \in \mathcal{D}_{k_j} \setminus \{0\}$  is satisfied for the mix-valued logical function

$$f_i \left( [X_k]_{k \in \mathcal{P}_i \setminus \{j\}}, (\nabla_{k_j, k_j}(U_{j,i}) \wedge_{k_j} X_j) \vee_{k_j} (\nabla_{k_j - a_j, k_j}(U_{j,i}) \wedge_{k_j} a_j) \right), \quad (18)$$

where  $\nabla_{k_j, k_j}$  and  $\nabla_{k_j - a_j, k_j}$  are defined in (1), and  $a_j$  is a parameter arbitrarily chosen from  $\mathcal{D}_{k_j}$ .

*Proof.* (Necessity) From Definition 2,  $f_i$  is independent of  $X_j$ . Now we prove that  $U_{j,i} \neq 0$  is satisfied for the function (18) by a reduction to absurdity. If  $U_{j,i} = 0$  and  $a_j \neq 0$ , then

$$(\nabla_{k_j, k_j}(U_{j,i}) \wedge_{k_j} X_j) \vee_{k_j} (\nabla_{k_j - a_j, k_j}(U_{j,i}) \wedge_{k_j} a_j) = ((k_j - 1) \wedge_{k_j} X_j) \vee_{k_j} (0 \wedge_{k_j} a_j) = X_j \vee_{k_j} 0 = X_j.$$

If  $U_{j,i} = 0$  and  $a_j = 0$ , then

$$\begin{aligned} & (\nabla_{k_j, k_j}(U_{j,i}) \wedge_{k_j} X_j) \vee_{k_j} (\nabla_{k_j - a_j, k_j}(U_{j,i}) \wedge_{k_j} a_j) \\ &= ((k_j - 1) \wedge_{k_j} X_j) \vee_{k_j} ((k_j - 1) \wedge_{k_j} 0) = X_j \vee_{k_j} 0 = X_j. \end{aligned}$$

Correspondingly,

$$f_i \left( [X_k]_{k \in \mathcal{P}_i \setminus \{j\}}, (\nabla_{k_j, k_j}(U_{j,i}) \wedge_{k_j} X_j) \vee_{k_j} (\nabla_{k_j - a_j, k_j}(U_{j,i}) \wedge_{k_j} a_j) \right) = f_i([X_k]_{k \in \mathcal{P}_i})$$

holds for  $U_{j,i} = 0$ . Therefore,  $f_i$  depends on  $X_j$ , which is a contradiction.

(Sufficiency) Assume that  $U_{j,i} = p$ ,  $p \in \mathcal{D}_{k_j} \setminus \{0, a_j\}$  is satisfied for the function (18). Then,

$$(\nabla_{k_j, k_j}(U_{j,i}) \wedge_{k_j} X_j) \vee_{k_j} (\nabla_{k_j - a_j, k_j}(U_{j,i}) \wedge_{k_j} a_j) = (0 \wedge_{k_j} X_j) \vee_{k_j} (0 \wedge_{k_j} a_j) = 0 \vee_{k_j} 0 = 0.$$

If  $a_j \neq 0$  and  $U_{j,i} = a_j$ , then it holds that

$$(\nabla_{k_j, k_j}(U_{j,i}) \wedge_{k_j} X_j) \vee_{k_j} (\nabla_{k_j - a_j, k_j}(U_{j,i}) \wedge_{k_j} a_j) = (0 \wedge_{k_j} X_j) \vee_{k_j} ((k_j - 1) \wedge_{k_j} a_j) = 0 \vee_{k_j} a_j = a_j.$$

Thus,  $f_i$  is independent of  $X_j$ .

We define the edge  $j \rightarrow i$  as  $e_{i,j} \in \mathcal{E}$ . Let  $s(e_{i,j}) = j$  and  $d(e_{i,j}) = i$  be the source and destination of edge  $e_{i,j}$ , respectively. We present the concept of (minimal) feedback arc set for a directed graph to remove as few edges as possible.

**Definition 3** ([34]). Given a directed graph, a feedback arc set is a subset of edges containing at least one edge of each directed cycle, and it is called a minimal feedback arc set if its cardinality is minimum.

However, it is NP-hard to exactly find a minimal feedback arc set [34]. Several effective algorithms in polynomial time have been proposed to identify a close-to-minimal feedback arc set, such as a Monte Carlo algorithm, whose computational complexity is  $O(n^3)$  [34]. Using the Monte Carlo algorithm, one can obtain a close-to-minimal feedback arc set in  $\mathcal{W} = \{\mathcal{N}, \mathcal{E}\}$ . Assume that  $\Xi = \{e_1, \dots, e_v\}$  is a close-to-minimal feedback arc set with cardinality  $v \in [1, |\mathcal{E}|]_{\mathbb{N}}$ .

To find the update functions that should be modified, we define

$$\tau := \bigcup_{i=1}^v \{d(e_i)\} = \{d_1, \dots, d_\kappa\}, \quad (19)$$

where  $\tau \subseteq \mathcal{N}$  and  $\kappa \leq v$ . From (19), update functions  $f_{d_r}$ ,  $r \in [1, \kappa]_{\mathbb{N}}$  should be modified via edge removal control.

The next step is to acquire the regulators that need to be controlled in update functions  $f_{d_r}$ ,  $r \in [1, \kappa]_{\mathbb{N}}$ . To this end, we construct the following sets:

$$\Gamma(d_r) := \{e \in \Xi : d(e) = d_r\}, \quad r \in [1, \kappa]_{\mathbb{N}}. \quad (20)$$

For any edge  $e \in \Gamma(d_r)$ , the destination of  $e$  is  $d_r$ . Let

$$\Gamma(d_r) = \{e_{d_r^1}, \dots, e_{d_r^{\alpha_r}}\}, \quad (21)$$

where  $d_r^1 < \dots < d_r^{\alpha_r}$ . Apparently,  $\sum_{r=1}^{\kappa} \alpha_r = v$ . Further, we define the source set of edges in  $\Gamma(d_r)$  as

$$S(d_r) = \bigcup_{i=d_r^1}^{d_r^{\alpha_r}} \{s(e_i)\} = \{\eta_r^1, \dots, \eta_r^{\alpha_r}\}. \quad (22)$$

If we remove all edges in  $\Xi$ , then the network graph of system (5) will be acyclic. Theorem 1 shows that system (5) has only a fixed point attractor. However, an unintended attractor other than  $X_e = (\xi_1, \dots, \xi_n)$  may be created via edge removal and the improper setting of  $a_j \in \mathcal{D}_{k_j}$ . Our final goal is to guarantee that  $X_e$  is still an attractor of system (5) after edge removal. This goal can be achieved by appropriately adjusting  $a_j \in \mathcal{D}_{k_j}$  in (18). We use the component of fixed point  $X_e$  to determine  $a_j = \xi_j$  and construct the following edge removal control:

$$U_{j,d_r} = \max\{1, \xi_j\}, \quad r \in [1, \kappa]_{\mathbb{N}}, \quad j \in S(d_r). \quad (23)$$

Then, we exert the edge removal control on system (5) and derive

$$X_i(t+1) = \begin{cases} \hat{f}_i([X_j(t), U_{j,i}]_{j \in \mathcal{P}_i}), & \text{if } \varphi_i(t+1) = 1 \text{ and } i \in \tau, \\ \hat{f}_i([X_j(t)]_{j \in \mathcal{P}_i}), & \text{if } \varphi_i(t+1) = 1 \text{ and } i \in \mathcal{N} \setminus \tau, \\ X_i(t), & \text{if } \varphi_i(t+1) = 0, \end{cases} \quad (24)$$

where  $\varphi_i(t)$  is the same as that in (6),  $\hat{f}_i = f_i$ ,  $i \in \mathcal{N} \setminus \tau$ , and

$$\begin{aligned} & \hat{f}_{d_r}([X_j(t), U_{j,d_r}]_{j \in \mathcal{P}_{d_r}}) \\ &= f_{d_r} \left( \left[ (\nabla_{k_j, k_j}(U_{j,d_r}) \wedge_{k_j} X_j(t)) \vee_{k_j} (\nabla_{k_j - \xi_j, k_j}(U_{j,d_r}) \wedge_{k_j} \xi_j) \right]_{j \in S(d_r)}, \right. \\ & \left. [X_k(t)]_{k \in \mathcal{P}_{d_r} \setminus S(d_r)} \right), \quad r \in [1, \kappa]_{\mathbb{N}}. \end{aligned} \quad (25)$$

$d_r$  and  $S(d_r)$  are given in (21) and (22), respectively. According to Theorem 1 and Proposition 2, we present the following result.

**Theorem 2.** Under the edge removal control (23), system (5) is finite-time stable at  $X_e = (\xi_1, \dots, \xi_n)$ .

*Proof.* If  $\xi_j = 0$ , then  $U_{j,d_r} = 1$  and

$$(\nabla_{k_j, k_j}(U_{j,d_r}) \wedge_{k_j} X_j) \vee_{k_j} (\nabla_{k_j - \xi_j, k_j}(U_{j,d_r}) \wedge_{k_j} \xi_j) = 0. \quad (26)$$

If  $\xi_j \neq 0$ , then  $U_{j,d_r} = \xi_j$  and

$$(\nabla_{k_j,k_j}(U_{j,d_r}) \wedge_{k_j} X_j) \vee_{k_j} (\nabla_{k_j-\xi_j,k_j}(U_{j,d_r}) \wedge_{k_j} \xi_j) = \xi_j. \quad (27)$$

According to Proposition 2, one concludes that edges  $j \rightarrow d_r$ ,  $j \in S(d_r)$ ,  $r \in [1, \kappa]_{\mathbb{N}}$  in the network graph  $\mathcal{W} = \{\mathcal{N}, \mathcal{E}\}$  are removed. According to the construction of  $d_r$  and  $S(d_r)$ , one obtains that

$$\bigcup_{r=1}^{\kappa} \{\eta_r^1 \rightarrow d_r, \dots, \eta_r^{\alpha_r} \rightarrow d_r\} = \Xi = \{e_1, \dots, e_v\},$$

which is a close-to-minimal feedback arc set of network graph  $\mathcal{W} = \{\mathcal{N}, \mathcal{E}\}$ . Thus, all edges in the close-to-minimal feedback arc set are removed. By Definition 3, the network graph  $\mathcal{W} = \{\mathcal{N}, \mathcal{E}\}$  of system (5) is acyclic. From Theorem 1, system (5) with the edge removal control (23) is finite-time stable, and thus has a fixed point as the unique attractor.

Now, we prove that  $X_e = (\xi_1, \dots, \xi_n)$  is the fixed point of system (5) after edge removal. One derives from  $\{X_e\} \in \Theta$  that  $f_i(X_e) = \xi_i$ ,  $\forall i \in \mathcal{N}$ . For any  $t \in \mathbb{N}$ , assume that  $X(t) = X_e$ . We divide  $\mathcal{N}$  into the following three disjoint sets:  $\mathcal{N}_1 = \{i : \varphi_i(t+1) = 1 \text{ and } i \in \tau\}$ ,  $\mathcal{N}_2 = \{i : \varphi_i(t+1) = 1 \text{ and } i \notin \tau\}$  and  $\mathcal{N}_3 = \{i : \varphi_i(t+1) = 0\}$ . If  $i \in \mathcal{N}_1$ , then it holds from (25)–(27) that

$$X_i(1; X(t)) = \hat{f}_i([\xi_j]_{j \in S(i)}, [X_k(t)]_{k \in \mathcal{P}_i \setminus S(i)}) = \hat{f}_i([\xi_j]_{j \in \mathcal{P}_i}) = \xi_i. \quad (28)$$

For  $i \in \mathcal{N}_2$ , one has  $X_i(1; X(t)) = \hat{f}_i([\xi_j]_{j \in \mathcal{P}_i}) = \xi_i$ . On the basis of (24), it is easy to obtain that  $X_i(1; X(t)) = X_i(t) = \xi_i$ ,  $i \in \mathcal{N}_3$ . Therefore,

$$X(1; X(t)) = X(t) = X_e, \forall t \in \mathbb{N}.$$

According to the definition of trap domain, one concludes that  $\mathcal{T}_d = \{X_e\}$  is a trap domain, which together with  $|\mathcal{T}_d| = 1$  implies that  $X_e$  is a fixed point. Thus, system (5) with the edge removal control (23) is finite-time stable at  $X_e$ .

Based on (19) and (22), we establish Algorithm 2 for designing edge removal control in (25).

---

**Algorithm 2** Edge removal control design.

---

**Require:**  $\Xi \Leftarrow \{e_1, \dots, e_v\}$ ,  $\tau \Leftarrow \emptyset$ ,  $S(i) \Leftarrow \emptyset$ ,  $f_i$ ,  $i \in \mathcal{N}$ ;

**Ensure:**  $\hat{f}_i$ ,  $i \in \mathcal{N}$ ;

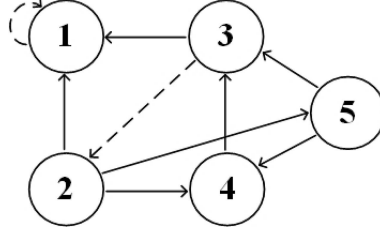
```

1: for  $i = 1, \dots, v$  do
2:    $\tau \Leftarrow \tau \cup \{d(e_i)\}$ ;
3: end for
4: for  $d_r \in \tau$  do
5:   for  $i = 1, \dots, v$  do
6:     if  $d(e_i) = d_r$  then
7:        $S(d_r) \Leftarrow S(d_r) \cup \{s(e_i)\}$ ;
8:     end if
9:   end for
10: end for
11: for  $i = 1, \dots, n$  do
12:   if  $i \in \tau$  then
13:     Calculate  $\hat{f}_i$  based on  $f_i$  and (25);
14:   else
15:      $\hat{f}_i \Leftarrow f_i$ ;
16:   end if
17: end for
```

---

**Remark 4.** The edge removal control design comprises two parts: finding a close-to-minimal feedback arc set [34] of the network graph and conducting Algorithm 2. Therefore, the computational complexity of designing edge removal control is  $O(n^3)$ . Comparatively, pinning control [26] is also a useful method to achieve the edge removal of DISs. However, in the extreme case where a certain node has  $n$  parent nodes, the computational complexity of the pinning control method is  $k_1 \times \dots \times k_n$ .

**Remark 5.** In [30], an edge removal control was introduced to delete the edges of BNs based on Boolean operators, which is thus not applicable to MLNs. Meanwhile, the proposed edge removal control is suitable for the dynamics of MLNs and stabilizes system (5) to the desired equilibrium based on parameter  $a_j \in \mathcal{D}_{k_j}$  and operators  $\nabla_{k_j,k_j}$  and  $\nabla_{k_j-a,k_j}$ . Moreover, it has wider applications than the edge removal control in [30].



**Figure 3** Network graph of finite-field network (29), where the dashed lines are the edges to be removed.

**Remark 6.** The main results of this study are derived based on a specified minimal feedback arc set. Notably, the choice of a minimal feedback arc set could influence not only computational efficiency but also the practical feasibility of implementing edge removal. Therefore, how to use both computational efficiency and biological significance as performance measures to optimize the selection of a minimal feedback arc set is an interesting problem, which is an important future research direction.

## 4 Illustrative examples

Let  $p$  be a prime number. Finite-field networks with  $n$  agents over  $\mathbb{F}_p := [0, p]_{\mathbb{N}}$  are effective in the investigation of multi-agent systems [3]. The network graph of a finite-field network is denoted by  $\mathcal{W} = \{\mathcal{N}, \mathcal{E}\}$ . The dynamics of the  $i$ -th agent is described as follows:

$$X_i(t+1) = a_{i,1} \times_p X_1(t) +_p \cdots +_p a_{i,n} \times_p X_n(t),$$

where  $a_{i,j} \in \mathbb{F}_p$  is the weight associated with the edge  $j \rightarrow i \in \mathcal{E}$ , and  $+_p$  and  $\times_p$  are modular addition and multiplication over field  $\mathbb{F}_p$ .

The consensus of finite-field networks has attracted considerable attention in some recent studies [3, 35–37], most of which are based on the network topology. When the network topology of a finite-field network cannot guarantee its consensus, we can introduce the edge removal control to achieve this goal.

**Example 2.** Consider the following finite-field network over  $\mathbb{F}_5 := [0, 4]_{\mathbb{N}}$ :

$$\begin{cases} X_1(t+1) = X_1(t) +_5 2 \times_5 X_2(t) +_5 3 \times_5 X_3(t), \\ X_2(t+1) = X_3(t), \\ X_3(t+1) = 4 \times_5 X_4(t) +_5 2 \times_5 X_5(t), \\ X_4(t+1) = 3 \times_5 X_2(t) +_5 3 \times_5 X_5(t), \\ X_5(t+1) = X_2(t). \end{cases} \quad (29)$$

Figure 3 shows the network graph of system (29).

Based on [3], one obtains that system (29) cannot achieve the finite-time consensus. We aim to design an edge removal control that allows it to achieve finite-time consensus.

Using the Monte Carlo algorithm, a close-to-minimal feedback arc set is  $\{1 \rightarrow 1, 3 \rightarrow 2\}$ , which should be removed in Figure 3. If let parameters  $a_1 = 0$ ,  $a_3 = 1$ , then based on Algorithm 2, we obtain the following new dynamics of system (29):

$$\begin{cases} X_1(t+1) = 2 \times_5 X_2(t) +_5 3 \times_5 X_3(t), \\ X_2(t+1) = 1, \\ X_3(t+1) = 4 \times_5 X_4(t) +_5 2 \times_5 X_5(t), \\ X_4(t+1) = 3 \times_5 X_2(t) +_5 3 \times_5 X_5(t), \\ X_5(t+1) = X_2(t). \end{cases} \quad (30)$$

Taking  $T^* = 4$ , one concludes that

$$X(t; X_0) = (0, 1, 1, 1, 1)^\top, \quad \forall t \geq T^*$$



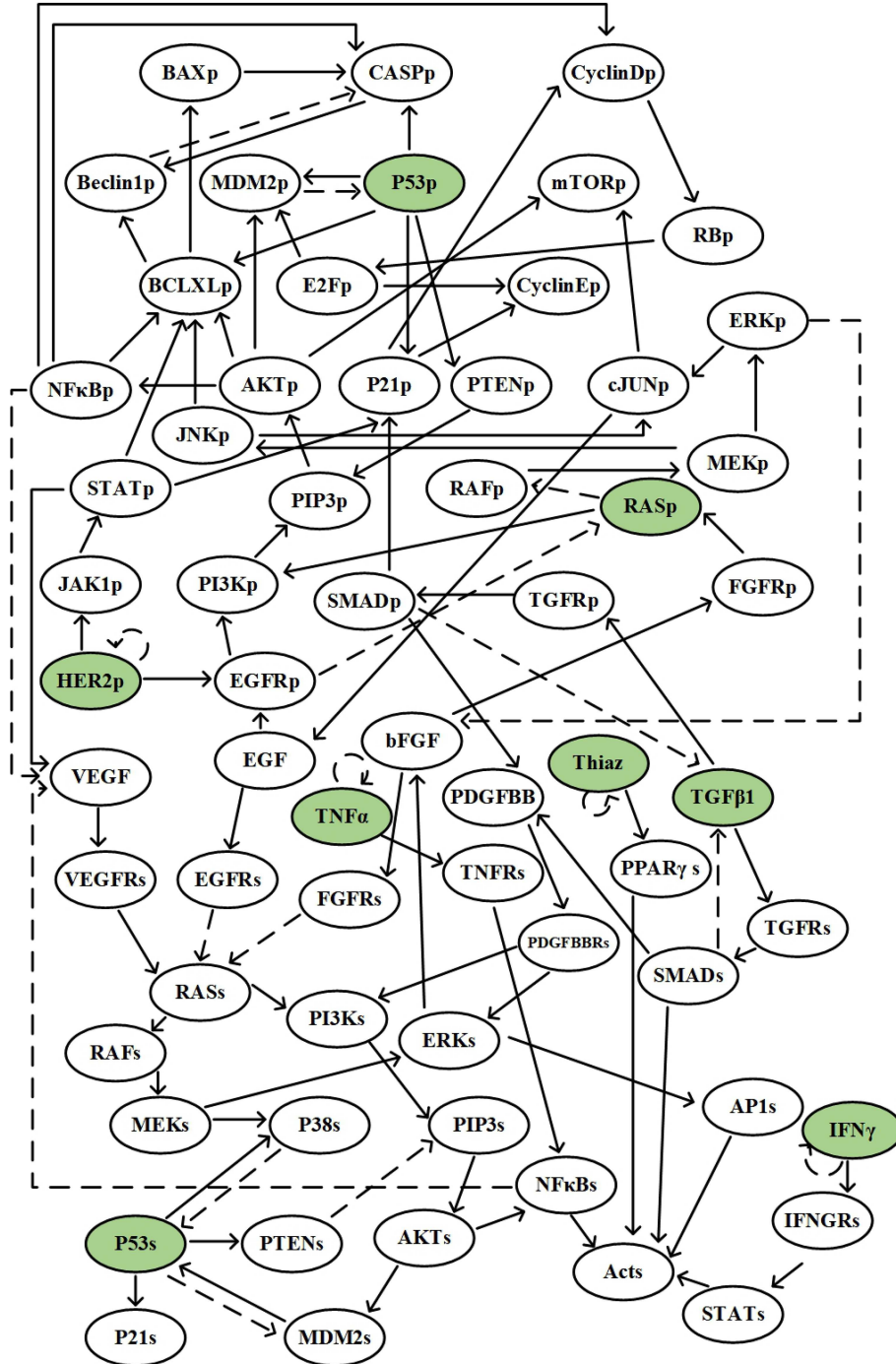
$$\left\{ \begin{array}{l}
 \text{VEGF: } X_1(t+1) = X_{22}(t) \vee X_{42}(t) \vee X_{47}(t), \text{ EGF: } X_2(t+1) = X_{52}(t), \\
 \text{bFGF: } X_3(t+1) = X_{29}(t) \vee X_{50}(t), \text{ TNF}\alpha: X_4(t+1) = X_4(t), \\
 \text{PDGFBB: } X_5(t+1) = X_{19}(t) \vee X_{41}(t), \text{ Thiaz: } X_6(t+1) = X_6(t), \\
 \text{TGF}\beta 1: X_7(t+1) = X_{19}(t) \vee X_{41}(t), \text{ IFN}\gamma: X_8(t+1) = X_8(t), \text{ VEGFRs: } X_9(t+1) = X_1(t), \\
 \text{EGFRs: } X_{10}(t+1) = X_2(t), \text{ FGFRs: } X_{11}(t+1) = X_3(t), \text{ TNFRs: } X_{12}(t+1) = X_4(t), \\
 \text{PDGFBBRs: } X_{13}(t+1) = X_5(t), \text{ PPAR}\gamma\text{s: } X_{14}(t+1) = X_6(t), \text{ TGF}\beta\text{Rs: } X_{15}(t+1) = X_7(t), \\
 \text{IFNGRs: } X_{16}(t+1) = X_8(t), \text{ RASs: } X_{17}(t+1) = X_9(t) \vee X_{10}(t) \vee X_{11}(t), \\
 \text{PI3Ks: } X_{18}(t+1) = X_{13}(t) \vee X_{17}(t), \text{ SMADs: } X_{19}(t+1) = X_{15}(t), \text{ STATs: } X_{20}(t+1) = X_{16}(t), \\
 \text{RAFs: } X_{21}(t+1) = X_{17}(t), \text{ NF}\kappa\text{Bs: } X_{22}(t+1) = X_{12}(t) \vee X_{28}(t), \text{ P38s: } X_{23}(t+1) = X_{24}(t) \vee X_{26}(t), \\
 \text{MEKs: } X_{24}(t+1) = X_{21}(t), \text{ PIP3s: } X_{25}(t+1) = (\neg X_{27}(t)) \wedge X_{18}(t), \\
 \text{P53s: } X_{26}(t+1) = (\neg X_{32}(t)) \wedge X_{23}(t), \text{ PTENS: } X_{27}(t+1) = X_{26}(t), \text{ AKTs: } X_{28}(t+1) = X_{25}(t), \\
 \text{ERKs: } X_{29}(t+1) = X_{13}(t) \vee X_{24}(t), \text{ AP1s: } X_{30}(t+1) = X_{29}(t), \text{ P21s: } X_{31}(t+1) = X_{26}(t), \\
 \text{MDM2s: } X_{32}(t+1) = X_{26}(t) \vee X_{28}(t), \\
 \text{Acts: } X_{33}(t+1) = X_{19}(t) \wedge (\neg X_{20}(t) \vee \neg X_{14}(t)) \wedge (X_{22}(t) \vee X_{30}(t)), \\
 \text{EGFRp: } X_{34}(t+1) = X_2(t) \vee X_{37}(t), \text{ FGFRp: } X_{35}(t+1) = X_3(t), \text{ TGF}\beta\text{Rp: } X_{36}(t+1) = X_7(t), \\
 \text{HER2p: } X_{37}(t+1) = X_{37}(t), \text{ JAK1p: } X_{38}(t+1) = X_{37}(t), \text{ PI3Kp: } X_{39}(t+1) = X_{34}(t) \vee X_{40}(t), \\
 \text{RASp: } X_{40}(t+1) = X_{34}(t) \vee X_{35}(t), \text{ SMADp: } X_{41}(t+1) = X_{36}(t), \text{ STATp: } X_{42}(t+1) = X_{38}(t), \\
 \text{PIP3p: } X_{43}(t+1) = \neg X_{49}(t) \wedge X_{39}(t), \text{ RAFp: } X_{44}(t+1) = X_{40}(t), \\
 \text{P21p: } X_{45}(t+1) = \neg X_{42}(t) \wedge (X_{41}(t) \vee X_{61}(t)), \text{ MEKp: } X_{46}(t+1) = X_{44}(t), \\
 \text{NF}\kappa\text{Bp: } X_{47}(t+1) = X_{48}(t), \text{ AKTp: } X_{48}(t+1) = X_{43}(t), \text{ PTENp: } X_{49}(t+1) = X_{61}(t), \\
 \text{ERKp: } X_{50}(t+1) = X_{46}(t), \text{ E2Fp: } X_{51}(t+1) = \neg X_{54}(t), \text{ cJUNp: } X_{52}(t+1) = X_{50}(t) \vee X_{56}(t), \\
 \text{CyclinDp: } X_{53}(t+1) = \neg X_{45}(t) \wedge X_{47}(t), \text{ RBp: } X_{54}(t+1) = \neg X_{53}(t), \\
 \text{BCLXLp: } X_{55}(t+1) = \neg X_{61}(t) \wedge (X_{47}(t) \vee X_{42}(t) \vee X_{48}(t) \vee X_{56}(t)), \text{ JNKp: } X_{56}(t+1) = X_{46}(t), \\
 \text{mTORp: } X_{57}(t+1) = \neg X_{52}(t) \wedge X_{48}(t), \text{ BAXp: } X_{58}(t+1) = \neg X_{55}(t), \\
 \text{Beclin1p: } X_{59}(t+1) = \neg X_{55}(t) \wedge \neg X_{63}(t), \text{ MDM2p: } X_{60}(t+1) = \neg X_{51}(t) \wedge (X_{48}(t) \vee X_{61}(t)), \\
 \text{P53p: } X_{61}(t+1) = \neg X_{60}(t), \text{ CyclinEp: } X_{62}(t+1) = \neg X_{45}(t) \wedge X_{51}(t), \\
 \text{CASPp: } X_{63}(t+1) = \neg X_{47}(t) \wedge (X_{58}(t) \vee X_{59}(t) \vee X_{61}(t)).
 \end{array} \right. \quad (33)$$

According to Algorithm 2, the step counts of finding a minimal feedback arc set in Figure 5 and designing the edge removal control are  $63^3 + 15 = 250062$ . Meanwhile, for the method in [26], which is used to find a minimal feedback arc set of the network graph and design the pinning control, the step counts are  $63^3 + (2^4)^3 + 2 \times (2^3)^3 + 9 \times (2^2)^3 + 3 \times 2^3 = 255767$ . Clearly, Algorithm 2 requires fewer steps, indicating that it can improve computational efficiency and speed.

**Remark 7.** As reported in [38], TNF $\alpha$ , Thiaz, TGF $\beta$ 1, IFN $\gamma$ , P53s, and P53p correlate with apoptotic cell death, and RASp and HER2p are proven to be important across most strategies. The edges removed in this example cover all eight genes and provide the theoretical support for these key genes in the dynamics of the pancreatic cancer network. Further, as reported in [39], silencing RASp–RAFp–MEKp interactions via CRISPR knockout may present an exploitable avenue for targeted treatments in the pancreatic cancer network, which supports the removal of edge “RASp–RAFp” in Figure 5. In addition, because CRISPR–Cas9-mediated gene knockout directly corresponds to permanent edge removal, removing the edge “RASp–RAFp” is irreversible.

## 5 Conclusion

In this study, we discussed the stability of DISs with DAU via the network graph. We proved that DISs with DAU are finite-time stable if the network graph is acyclic and constructed an algorithm for seeking out the equilibrium in



**Figure 5** (Color online) Network graph of the pancreatic cancer network, where the dashed lines are the edges to be removed.

polynomial time. Then, based on the feedback arc set of the network graph, we designed an edge removal control to achieve the stability of DISs with DAU and adjusted several parameters to reach the desired equilibrium. Moreover, we showed that the computational complexity of the stability criterion and control scheme is of polynomial time. Finally, we used a finite-field network and a practical gene regulatory network to show the effectiveness of the proposed edge removal control. Future studies will explore the consensus theory [40] of finite-field networks with DAU based on edge removal control.

In practical situations, when there are residual cycles in a network graph, analyzing the finite-time stability of a DIS with DAU remains a very challenging problem. The theory of positive and negative cycles was initially introduced in [19] to explore the non-oscillation of DISs. This theory can be employed to analyze the finite-time stability of DISs with near-cyclic or weakly cyclic structures.

In real-world gene regulatory networks, silencing interactions involving some target genes may disrupt oncogenic signaling or even affect non-target pathways. Therefore, combining our theoretical framework with domain-specific constraints would be a feasible strategy for enhancing the applicability of our theoretical findings. Future work can explore how edge removal control balances the therapeutic efficacy with off-target effects by integrating the domain-specific constraints into our feedback arc set selection process.

**Acknowledgements** The work was supported by National Natural Science Foundation of China (Grant Nos. 62473239, 62073202), Major Basic Research Project of Natural Science Foundation of Shandong Province (Grant No. ZR2024ZD41), and Scientific Research Innovation Capability Support Project for Young Faculty (Grant No. SRICSPYF-BS2025004).

## References

- 1 Park J C, Jang S Y, Lee D, et al. A logical network-based drug-screening platform for Alzheimer's disease representing pathological features of human brain organoids. *Nat Commun*, 2021, 12: 280
- 2 Cheng D Z, Li C X. Design of zero-determinant strategies and its application to networked repeated games. *Sci China Inf Sci*, 2024, 67: 202201
- 3 Pasqualetti F, Borra D, Bullo F. Consensus networks over finite fields. *Automatica*, 2014, 50: 349–358
- 4 Yan Y Y, Cheng D Z, Feng J-E, et al. Survey on applications of algebraic state space theory of logical systems to finite state machines. *Sci China Inf Sci*, 2023, 66: 111201
- 5 Li H T, Liu Z Q, Li W R. Nonsingularity of grain-like cascade feedback shift registers subject to fault attacks. *Sci China Inf Sci*, 2024, 67: 192203
- 6 Qasem O, Gao W, Vamvoudakis K G. Adaptive optimal control of continuous-time nonlinear affine systems via hybrid iteration. *Automatica*, 2023, 157: 111261
- 7 Robert F. *Discrete Iterations: A Metric Study*. New York: Springer, 2012
- 8 Cheng D, Qi H, Li Z. *Analysis and Control of Boolean Networks: A Semi-Tensor Product Approach*. London: Springer, 2011
- 9 Veliz-Cuba A, Jarrah A S, Laubenbacher R. Polynomial algebra of discrete models in systems biology. *Bioinformatics*, 2010, 26: 1637–1643
- 10 Zhang L, Feng J, Feng X, et al. Further results on disturbance decoupling of mix-valued logical networks. *IEEE Trans Automat Contr*, 2013, 59: 1630–1634
- 11 Cheng D Z, Zhang X. Semi-tensor product-based convolutional neural network. *Sci China Inf Sci*, 2026, 69: 140201
- 12 Zhu S, Lu J, Ho D W C, et al. Minimal control nodes for strong structural observability of discrete-time iterative systems: explicit formulas and polynomial-time algorithms. *IEEE Trans Automat Contr*, 2024, 69: 2158–2173
- 13 Li W R, Li H T. Non-oscillation of asynchronous stochastic logical networks through negative circuits of the joint interaction graph. *Sci China Inf Sci*, 2026, 69: 140205
- 14 Zhao R, Wang C X, Yu Y Y, et al. Fault detectability of Boolean control networks via nonaugmented methods. *Sci China Inf Sci*, 2023, 66: 222205
- 15 Zhong J, Lu J, Huang T, et al. Controllability and synchronization analysis of identical-hierarchy mixed-valued logical control networks. *IEEE Trans Cybern*, 2017, 47: 3482–3493
- 16 Saadatpour A, Albert I, Albert R. Attractor analysis of asynchronous Boolean models of signal transduction networks. *J Theor Biol*, 2010, 266: 641–656
- 17 Chaves M, Sontag E D, Albert R. Methods of robustness analysis for Boolean models of gene control networks. *IEE Proc Syst Biol*, 2006, 153: 154–167
- 18 Kim J, Park S M, Cho K H. Discovery of a kernel for controlling biomolecular regulatory networks. *Sci Rep*, 2013, 3: 2223
- 19 Richard A. Negative circuits and sustained oscillations in asynchronous automata networks. *Adv Appl Math*, 2010, 44: 378–392
- 20 Richard A. Positive and negative cycles in Boolean networks. *J Theor Biol*, 2019, 463: 67–76
- 21 Su C, Pang J. Target control of asynchronous Boolean networks. *IEEE ACM Trans Comput Biol Bioinf*, 2023, 20: 707–719
- 22 Li Y, Li H. Controllability and stabilization of periodic switched Boolean control networks with application to asynchronous updating. *Nonlinear Anal-Hybrid Syst*, 2021, 41: 101054
- 23 Tong L, Liang J. Fault detectability of asynchronous delayed Boolean control networks with sampled-data control. *IEEE Trans Netw Sci Eng*, 2024, 11: 724–735
- 24 Dennunzio A, Formenti E, Manzoni L, et al. Computational complexity of finite asynchronous cellular automata. *Theor Comput Sci*, 2017, 664: 131–143
- 25 Soh T, Magnin M, Le B D, et al. SAF: SAT-based attractor finder in asynchronous automata networks. In: *Proceedings of CMSB*, 2023. 14137: 175–183
- 26 Zhong J, Pan Q, Xu W, et al. A framework of pinning control for nonperiodical stable behaviors of large-scale asynchronous Boolean networks. *IEEE Trans Automat Contr*, 2024, 69: 5711–5726
- 27 Zhong Q, Simonis N, Li Q, et al. Edgetic perturbation models of human inherited disorders. *Mol Syst Biol*, 2009, 5: 321
- 28 Campbell C, Albert R. Edgetic perturbations to eliminate fixed-point attractors in Boolean regulatory networks. *Chaos-An Interdisciplinary J Nonlinear Sci*, 2019, 29: 023130
- 29 Rashid M, Hari K, Thampi J, et al. Network topology metrics explaining enrichment of hybrid epithelial/mesenchymal phenotypes in metastasis. *PLoS Comput Biol*, 2022, 18: e1010687
- 30 Li W, Li H, Yang X. Edge removal towards asymptotical stabilizability of Boolean networks under asynchronous stochastic update. *Syst Control Lett*, 2023, 181: 105639
- 31 Xu X, Hong Y. Matrix approach to model matching of asynchronous sequential machines. *IEEE Trans Automat Contr*, 2013, 58: 2974–2979
- 32 Diekert V, Muscholl A. Deterministic asynchronous automata for infinite traces. *Acta Informatica*, 1994, 31: 379–397
- 33 Richard A, Comet J P. Necessary conditions for multistationarity in discrete dynamical systems. *Discrete Appl Math*, 2007, 155: 2403–2413
- 34 Kudelić R, Ivković N. Ant inspired Monte Carlo algorithm for minimum feedback arc set. *Expert Syst Appl*, 2019, 122: 108–117
- 35 Li X, Chen M Z Q, Su H, et al. Consensus networks with switching topology and time-delays over finite fields. *Automatica*, 2016, 68: 39–43
- 36 Li Y, Li H, Ding X, et al. Leader-follower consensus of multiagent systems with time delays over finite fields. *IEEE Trans Cybern*, 2019, 49: 3203–3208
- 37 Lin L, Cao J, Lam J, et al. Leader-follower consensus over finite fields. *IEEE Trans Automat Contr*, 2024, 69: 4718–4725
- 38 Plaughter D, Murrugarra D. Modeling the pancreatic cancer microenvironment in search of control targets. *Bull Math Biol*, 2021, 83: 1–26
- 39 Szlachta K, Kuscic C, Tufan T, et al. CRISPR knockout screening identifies combinatorial drug targets in pancreatic cancer and models cellular drug response. *Nat Commun*, 2018, 9: 4275
- 40 Zhang R C, Liu Z X, Chen G, et al. Minimum number of prejudiced agents needed for consensus of weighted median opinion dynamics. *Sci China Inf Sci*, 2025, 68: 210207