A New Approach to Pinning Control of Boolean Networks

Jie Zhong, Daniel W.C. Ho, *Follow, IEEE* and Jianquan Lu*, *Senoir Member, IEEE*

arXiv:1912.01411v2 [eess.SY] 31 Oct 2020

Abstract—Boolean networks (BNs) are discrete-time systems where nodes are inter-connected (here we call such connection rule among nodes as network structure), and the dynamics of each gene node is determined by logical functions. In this paper, we propose a new approach on pinning control design for global stabilization of BNs based on BNs' network structure, named as network-structure-based distributed pinning control. By deleting the minimum number of edges, the network structure becomes acyclic. Then, an efficient distributed pinning control is designed to achieve global stabilization. Compared with existing literature, the design of pinning control is not based on the state transition matrix of BNs. Hence, the computational complexity in this paper is reduced from $O(2^n \times 2^n)$ to $O(2 \times 2^K)$, where n is the number of nodes and $K \leq n$ is the largest number of in-neighbors of nodes. In addition, without using state transition matrix, global state information is no longer needed, the design of pinning control is just based on neighbors' local information, which is easier to be implemented. The proposed method is well demonstrated by several biological networks with different sizes. The results are shown to be simple and concise, while the traditional pinning control can not be applied for BNs with such a large dimension.

Index Terms—Boolean networks, Global stabilization, Distributed pinning control, Network structure, Semi-tensor product of matrices.

I. INTRODUCTION

In genetic regulatory networks, the behavior of genes or cells generates a logical phenomenon of activities, like active/inactive, expressed/unexpressed. As a typical logical system, Boolean network (BN) was first proposed by Kauffman to discover the internal behaviors of genes in genetic regulatory networks [1]. Since then, investigations on Boolean networks (BNs) have been a hot-spot to explore evolution patterns and structures [2]. In a BN each node representing a cell or a gene takes a value from two logical variables 1 and 0. In addition, the state evolution of each node is determined by a logical function described by logical operators and its neighboring nodes. This implies that the dynamical behavior of a BN is determined by both network structure describing the neighbor relationship between gene nodes and also by the

* Corresponding author: Jianquan Lu

Jie Zhong is with the College of Mathematics and Computer Science, Zhejiang Normal University, Jinhua 321004, China (e-mail: zhongjie0615@gmail.com).

Daniel W.C. Ho is with the Department of Mathematics, City University of Hong Kong, Kowloon, Hong Kong (e-email: madaniel@cityu.edu.hk).

Jianquan Lu is with the School of Mathematics, Southeast University, Nanjing 210096, China (e-mail: jqluma@seu.edu.cn)

value update logical functions. Since then, BNs have attracted great attention such as in the study of the oscillations of signal transduction networks [3] and identifying BNs' structure [4]–[6]. In addition, many fundamental results have been obtained to disclose the relationships between network structure of a BN and its fixed points and cyclic attractors [7].

Recently, a new matrix product called semi-tensor product (STP) approach was proposed in [8], which brings a new direction for further study of BNs. The STP of matrices is an extension of the conventional matrix product, and breaks the traditional dimension matching condition of matrix product. It was firstly proposed by Cheng and his colleagues [8] as a convenient tool to analyze logical functions, BNs and other finite-valued systems. Based on a bijective equivalence between logical variables and Boolean vectors, any logical function can be expressed in algebraic form. Under the framework of algebraic forms, a great deal of results have been established in the past few decades [9]–[12]. Some fundamental problems concerning Boolean (control) networks have been widely addressed, including probabilistic Boolean networks [13], observability [14], stability [15]–[18].

In biological systems or genetic networks, it is important to design therapeutic interventions that steer patients to desirable states, such as a healthy one, and maintain this state afterward [19]. For example, some kinds of control actions (node and edge deletions) have been proposed to identify control targets in some biological networks, such as p53-mdm2 network and T cell lymphocyte granular leukemia survival signaling network [20]. In [20], node deletion can model the control action of knockout of a node, while edge deletion can model the action of a drug that inactivates the corresponding interaction among two gene products.

Over the past years, stability and stabilization of BNs have attracted much attention. For example, stabilization of Boolean control networks (BCNs) based on two kinds of controls (open-loop control and state feedback control) has been addressed in [15]. Later, the results have been extended to the issue of stabilization to some periodic cycles [21]. In [16], Li *et al.* firstly proposed an approach to design state feedback stabilization for BCNs, and subsequently Li *et al.* presented a new way to design all possible feedback stabilizers in [22]. However, in these related papers, controllers are imposed to all nodes or applied to some nodes of BCNs. Due to the introduction of pinning control technique [23], [24], stabilization of BCNs under pinning controllers has been a new research direction [25], [26]. For example, in [23], stabilization under pinning controllers has been studied, and an algorithm to design pinning controllers is firstly presented based on the

state transition matrix of BNs.

Given a BN with n nodes, using STP method, one can obtain its algebraic form: $x(t+1) = Lx(t)$ (matrix L is called the state transition matrix with dimension $2^n \times 2^n$). According to the traditional pinning control method based on the state transition matrix L (named as L-based pinning control) proposed in [23], L-based pinning control can be designed to achieve stabilization by changing columns of matrix L. This implies that the information of matrix L is always needed in [23], and one cannot design pinning controllers without matrix L. Most of the results concerning the design of pinning controllers are based on this method; see [23]–[28]. However, the dimension of matrix L grows dramatically with the number of nodes. Thus, traditional L-based pinning control method seems to be hard for implementation on some large-scale genetic networks, which is one of the main drawbacks. In addition, since L-based pinning control method is based on the information of matrix L, the pinning controllers are determined based on global state information of all the nodes. Traditional L-based pinning control design is in the form of $u(t) = g(x_1(t), \ldots, x_n(t))$, where $x_1(t), \ldots, x_n(t)$ are global states of a BN. This is another drawback of traditional L-based pinning control design, which will lead to a high dimensional controller design. To the best of our knowledge, there is no result available for studying stabilization of BNs based on network structure of BNs and using neighbors' local information.

In our previous paper [24], the goal was to study the controllability problem of BCNs where some specific nodes are selected to be controlled. However, [24] does not present a method to determine the pinning controlled nodes and design the corresponding controllers. In addition, the controllability criteria in [24] are based on matrix L, which leads to an extremely high computational complexity and a very complex form for controllers. Moreover, in [24], the controllers are free control sequences which can take arbitrary values, but not state feedback controllers. In this paper, a new approach to design state feedback pinning control is proposed based on network structure of BNs without using the traditional state transition matrix L. An efficient and low dimensional control strategy is proposed based on local neighbors of controlled nodes. Motivated by the above discussions, this paper makes the following contributions:

(I) A new method named network-structure-based (NS-based) distributed pinning control is firstly proposed to achieve global stabilization. The design of NS-based distributed pinning control is based on the network structure describing the coupling connections among nodes, and the neighbors' local information. Compared with existing results on traditional L-based pinning control, the information of matrix L is no longer needed;

(II) The computational complexity will be dramatically reduced from $O(2^n \times 2^n)$ to $O(2 \times 2^K)$, where K is the largest number of in-neighbors of a node. The method of designing NS-based distributed pinning control can be applied for some large-scale BNs, and this is well illustrated by several biological networks with different network sizes like 90 nodes;

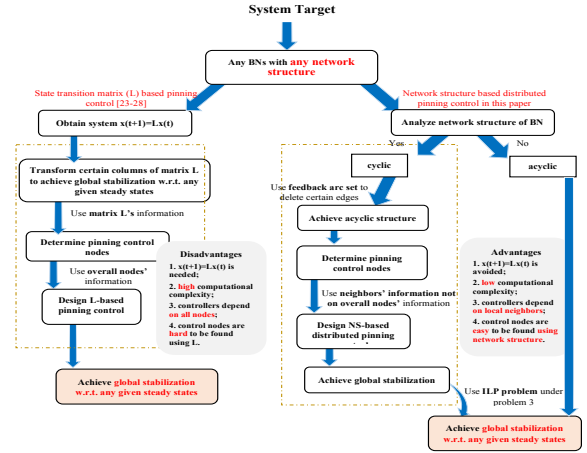(III) By searching feedback arc sets in the network structure

Fig. 1. The summarized comparisons between NS-based distributed pinning control and traditional L-based pinning control.

of BNs, pinning controlled nodes are easily determined to achieve global stability. Then, an NS-based distributed pinning control is designed based on the neighbors' local information but not on global state information. This NS-based distributed pinning leads to a lower dimensional controller design and be easier for implementation than traditional methods for designing controllers;

The summarized comparisons between NS-based distributed pinning control and traditional L-based pinning control can be observed from Fig. 1, which also implies one challenging topic of combining network structure with state transition digraph for designing efficient pinning control.

The remainder of this paper is presented as follows: In Section II, network structure and global stability of BNs are introduced. In Section III, the main results on NS-based distributed pinning control design are presented. In Section IV, two biological networks are given to demonstrate the validity of the results. A brief conclusion is given in Section V.

Some basic notations are given as follows: $\mathbf{1}_n = (1,1,\ldots,1)^\top$; $\mathscr{D} = \{0,1\}$: the logical domain; $\mathbb{N}^+$: the set of positive integers; $\mathscr{R}_{m \times n}$: the set of $m \times n$ real matrices; $|S|$ is the cardinal number of a given set S; $[a,b] = \{a, a+1, \ldots, b\}$, where $a < b$ and $a, b \in \mathbb{N}^+$; δ_n^i: the i-th column of the identity matrix I_n; $\Delta_n = \{\delta_n^1, \ldots, \delta_n^n\}$ denotes the columns set of I_n; $\mathrm{Col}_i(A)$: the i-th column of matrix A. $\mathrm{Col}(A)$ denotes the set of columns of A; $L = [\delta_n^{i_1}, \ldots, \delta_n^{i_s}]$ is called a logical matrix, and we simply denote it by $L = \delta_n[i_1, \ldots, i_s]$; $\mathscr{L}_{m \times n}$: the set of logical matrices with dimension $m \times n$; $\lfloor a \rfloor_S$ be the set of positive integers, $\lfloor a \rfloor_S = \{j : 0 < j < a, j \in S\}$, where S is a set of positive integers; Given variables $x_1, \ldots, x_n$ and an index set $I = \{i_1, \ldots, i_m\} \subseteq [1,n]$, define $[x_i]_{i \in I} = \{x_{i_1}, \ldots, x_{i_m}\}$; Swap matrix $W_{[m,n]} = [I_n \otimes \delta_m^1, I_n \otimes \delta_m^2, \ldots, I_n \otimes \delta_m^m]$; Power-reducing matrix $\Phi_{2^n} = \delta_{2^{2n}}[1, 2^n+2, (3-1)2^n+3, (4-1)2^n+4, \ldots, (2^n-2)2^n+2^n-1, 2^{2n}]$.

II. NETWORK STRUCTURE AND GLOBAL STABILITY

STP of matrices is a generalization of the conventional matrix product, which deals with the case when the dimension-matching condition of matrix product is not satisfied [8].

Definition 2.1 (STP): [8] Given two matrices $A \in \mathscr{R}_{n \times m}$ and $B \in \mathscr{R}_{p \times q}$, the STP of A and B, denoted by $A \ltimes B$, is defined as: $A \ltimes B = (A \otimes I_{l/m})(B \otimes I_{l/p})$, where l is the least common multiple of m and p, $\otimes$ is the Kronecker product of matrices.

In order to facilitate the use of the STP method in BNs, identify "1" and "0" with vectors, $1 \sim \delta_2^1$, $0 \sim \delta_2^2$, respectively. Under this framework, the equivalent algebraic form for any logical function can be obtained.

Lemma 2.1: [8] Let $f(a_1, \ldots, a_n) : (\mathscr{D})^n \to \mathscr{D}$ be a logical function. Then for every $(a_1, \ldots, a_n) \in (\Delta_2)^n$, there exists a unique matrix $F \in \mathscr{L}_{2 \times 2^n}$ such that $f(a_1, \ldots, a_n) = F \ltimes a_1 \ltimes \cdots \ltimes a_n$. F is called the structure matrix of f.

A Boolean function $f : \mathscr{D}^n \to \mathscr{D}$ consisting of n Boolean variables $x_1, \ldots, x_n \in \mathscr{D}$ is a discrete-time finite state dynamical system. Then, a BN can be described by a set of Boolean functions $f_1, \ldots, f_n$: $x_i(t+1) = f_i(x_1(t), \ldots, x_n(t)), i \in [1, n]$. Let $x(t) = (x_1(t), \ldots, x_n(t)) \in \mathscr{D}^n$ be the state at time t.

Note that using Boolean algebra, some variables may be nonfunctional in a logical function. For example, if $f(x_1, x_2) = (x_1 \wedge x_2) \vee (x_1 \wedge \neg x_2)$, then f is not dependent on variable x_2, because $(x_1 \wedge x_2) \vee (x_1 \wedge \neg x_2) = x_1$; while if $f(x_1, x_2) = x_1 \vee x_2$, then function f is dependent on variables x_1 and x_2. Then, we introduce the dependency of logical functions on variables.

Definition 2.2 (Functional Variables): A logical function $f(x_1, \ldots, x_i, \ldots, x_n) : \mathscr{D}^n \to \mathscr{D}$ is said to be dependent on variable x_i if there exists a tuple $\bar{x} \in \mathscr{D}^{n-1}$ such that $f(\bar{x}, x_i) \neq f(\bar{x}, \neg x_i)$, where $\bar{x} = (x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n)$. Variable x_i is called a functional variable of $f(\ldots, x_i, \ldots)$, otherwise, it is called a nonfunctional variable. In addition, if there is no non-functional variables, then f is called minimally represented.

Based on logical functions $f_1, \ldots, f_n$, introduce the interaction digraph describing the coupling connections among nodes of a BN which is called network structure.

Definition 2.3 (Interaction Digraph of BNs): Consider a BN with logical functions $f = (f_1, \ldots, f_n)$. The interaction digraph is a digraph denoted by $G_{id} = (V, E)$ of n vertices, the vertices set is $V = \{1, \ldots, n\}$. An edge $i \to j$ (simplified by $e_{ij} \in E$) exists in $G_{id} = (V, E)$ if and only if f_j is dependent on x_i.

Since a Boolean function $f : \mathscr{D}^n \to \mathscr{D}$ can be minimally represented, in this paper the following minimally represented BN is considered:

$$x_i(t+1) = f_i([x_j(t)]_{j \in \mathscr{N}_i}), \quad i \in [1, n]. \tag{1}$$

This implies that for each $i \in [1, n]$, all of the variables x_j, $j \in \mathscr{N}_i$ are functional variables for function f_i. Here, $x_i \in \mathscr{D}$, $i \in [1, n]$, is the i-th node of system (1), $\mathscr{N}_i \subseteq [1, n], i \in [1, n]$, is the index sets of in-neighbors of node x_i expressing the adjacency relation of nodes, $f_i : \mathscr{D}^{|\mathscr{N}_i|} \to \mathscr{D}, i \in [1, n]$, is Boolean function.

Then, the definitions of fixed points and cyclic attractors of system (1) are introduced as follows:

Definition 2.4 (Fixed Points and Cyclic Attractors): Given a BN (1), $\chi \in \mathscr{D}^n$ is called a fixed point if $L\chi = \chi$, and the sequence of states $\{\delta_{2^n}^{r_0}, \delta_{2^n}^{r_1}, \ldots, \delta_{2^n}^{r_{l-1}}\} \subseteq \Delta_{2^n}$ is called a cyclic attractor with length l if $L^l \delta_{2^n}^{r_0} = \delta_{2^n}^{r_0}$, the elements in $\{\delta_{2^n}^{r_0}, \delta_{2^n}^{r_1}, \ldots, \delta_{2^n}^{r_{l-1}}\}$ are pairwise distinct.

According to Lemma 2.1, suppose that the structure matrices of f_i are $A_i \in \mathscr{L}_{2 \times 2^{|\mathscr{N}_i|}}, i \in [1, n]$, then one can obtain the following algebraic form,

$$x_i(t+1) = A_i \ltimes_{j \in \mathscr{N}_i} x_j(t), \quad i \in [1, n]. \tag{2}$$

Suppose that for the set of in-neighbors of node x_i, $\mathscr{N}_i = \{b_j^i \mid b_j^i \in [1, n], j \in [1, |\mathscr{N}_i|]\}$. For the indexes $b_j^i \in [1, n]$, $j \in [1, |\mathscr{N}_i|]$, we assume that $1 \leq b_1^i < b_2^i < \cdots < b_{|\mathscr{N}_i|}^i \leq n$. For simplicity, we always assume that $\ltimes_{j \in \mathscr{N}_i} x_j(t) \triangleq x_{b_1^i}(t) \ltimes x_{b_2^i}(t) \ltimes \cdots \ltimes x_{b_{|\mathscr{N}_i|}^i}(t)$. For example, let $\mathscr{N}_i = \{3, 5, 6\}$, under the ordering assumption, then $\ltimes_{j \in \mathscr{N}_i} x_j(t) \triangleq x_3(t) x_5(t) x_6(t)$.

Further, letting $x(t) = \ltimes_{j=1}^n x_j(t) \in \Delta_{2^n}$, one can obtain the equivalent representation,

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

where $L \in \mathscr{L}_{2^n \times 2^n}$ is called the state transition matrix of system (3). Hence, according to matrix L, one can obtain the state transition graph of BN (3) consisting of 2^n nodes, i.e. $\{\delta_{2^n}^1, \delta_{2^n}^2, \ldots, \delta_{2^n}^{2^n}\}$, and there exists an edge from $\delta_{2^n}^i$ to $\delta_{2^n}^j$ in the state transition graph if and only if $L\delta_{2^n}^i = \delta_{2^n}^j$. The detailed calculations for the state transition matrix L can be obtained by using the STP method [8].

Remark 2.1: Note that the dimension of matrix $L \in \mathscr{L}_{2^n \times 2^n}$ grows exponentially with the size of network (depending on the number of nodes). Thus, all the algorithms based on matrix L have exponential complexity, which is a big noteworthy drawback [29]. As one can see from the discussions and analysis below, the design of NS-based distributed pinning controllers is not based on matrix L but on $A_i \in \mathscr{L}_{2 \times 2^{|\mathscr{N}_i|}}$ in (2). Generally, $|\mathscr{N}_i| \ll n$ in most gene regulatory networks, thus, the approach proposed in this paper reduces calculations and the computational complexity from $O(2^n \times 2^n)$ to $O(2 \times 2^K)$, where $K \leq n$ is the largest number of in-neighbors of nodes.

Based on the algebraic form (3), some results about global stability of BNs are recalled below.

Definition 2.5 (Global Stability): [8] [30] System (1) is said to be globally stable if there exists a unique fixed point as the attractor with no other cyclic attractors. In prarticular, system (1) is globally stable to $x^* \in \Delta_{2^n}$, if for any initial state $x_0 \in \Delta_{2^n}$, there exists an integer T, such that $x(t, x_0) = x^*, t \geq T$.

Lemma 2.2: [8] [30] System (1) is globally stable if the interaction digraph $G_{id} = (V, E)$ is acyclic. In addition, assume that stable state is $x^* = \delta_{2^n}^\gamma$, $\gamma \in [1, 2^n]$, then system (1) is globally stable to $x^* = \delta_{2^n}^\gamma$ if and only if there exists an integer T, such that $\text{Col}(L^T) = \{\delta_{2^n}^\gamma\}$.

III. MAIN RESULTS OF NS-BASED DISTRIBUTED PINNING CONTROL

Now, in this section, a new approach to design an NS-based distributed pinning control for global stabilization is proposed, based on the transformation of certain structure matrices of several nodes of BN (1), under which the interaction digraph $G_{id} = (V, E)$ will become acyclic. Hence, two questions should be answered in order to achieve global stabilization to a given fixed point, one is how to determine the pinning controlled nodes, another one is how to transform the structure matrices of the controlled nodes. Then, the following two main steps are introduced to answer the above two questions for designing

NS-based distributed pinning control, here we assume that the given fixed point is $\delta_{2^n}^{\gamma}$, $\gamma \in [1, 2^n]$:

(1) use the concept of feedback arc set to determine the controlled nodes (see Problem 1 below), and delete the minimum number of edges in the interaction digraph of BN (1) based on the transformation of structure matrices of controlled nodes (see Problem 2 below), such that $G_{id} = (V, E)$ becomes acyclic, which guarantees global stabilization but not to the given fixed point;

(2) after guaranteeing global stabilization, use an integer linear programming problem (see Problem 3 below) to further determine controlled nodes that are needed to achieve the given fixed point $\delta_{2^n}^{\gamma}$, and further transform the structure matrices $A_1, \ldots, A_n$, so that state $\delta_{2^n}^{\gamma}$ is the unique fixed point.

It should be noted that the control action of deleting edges of interaction digraph $G_{id} = (V, E)$ is plausible in genetic networks. In [20], two types of control actions (one is deletion of edges and another one is deletion of nodes) have been applied to Boolean molecular networks. For example, the deletion of an edge can be achieved by the use of therapeutic drugs to control some specific gene interactions. In the following subsections, we will use these two steps to design an NS-based distributed pinning control to achieve global stabilization to any given fixed point based on the interaction digraph $G_{id} = (V, E)$ of BN (1).

A. Step 1: guarantee global stabilization

Now, we consider the first main step, that is to delete the minimum number of edges in the interaction digraph $G_{id} = (V, E)$, such that the reduced digraph is acyclic. By using the depth-first search algorithm [31], one can obtain the cycles and fixed points in the digraph $G_{id} = (V, E)$. Firstly, we present some notations. Given an edge $e \in E$ in $G_{id} = (V, E)$, let $O_-(e)$ be the starting vertex and $O_+(e)$ be the ending vertex of the edge $e \in E$. Then, the concept of (minimum) feedback arc set for a given digraph is introduced in order to determine the pinning controlled nodes.

Definition 3.1 (Feedback Arc Set): [32] Given a directed network, a feedback arc set is a subset of edges containing at least one edge of every cycle, and it is called a minimum feedback arc set if its cardinality is minimum. Therefore, the removal of a feedback arc set renders the network acyclic.

Using the depth-first search algorithm, one can obtain all of the minimal feedback arc sets in the interaction digraph $G_{id} = (V, E)$, based on which, the following problem is considered to determine the pinning controlled nodes.

Problem 1 (Determination of Pinning Nodes): Assume that $\Omega_1 = \{e_1^1, \ldots, e_{c_2}^1\}, \ldots, \Omega_\kappa = \{e_1^\kappa, \ldots, e_{c_2}^\kappa\}$ are all of the minimal feedback arc sets with cardinality $c_2 \in [1, |E|]$ in the interaction digraph $G_{id} = (V, E)$. Here, $e_1^1, \ldots, e_{c_2}^1, \ldots, e_1^\kappa, \ldots, e_{c_2}^\kappa \in E$, $\kappa \in \mathbb{N}^+$. Then, minimize the cost function

$$c_1 \triangleq \min\{|\pi_j| : j = 1, \ldots, \kappa\}, \quad (4)$$

subject to conditions

$$\pi_j \triangleq \bigcup_{i=1}^{c_2} O_+(e_i^j), j \in [1, \kappa]. \quad (5)$$

Remark 3.1: As one can see from Problem 1, (4) and (5) are presented to find the minimum number of ending vertices among the edges in the minimal feedback arc sets, such that the digraph $G_{id} = (V, E)$ becomes acyclic if the edges in the minimal feedback arc sets are deleted. During the past few years, much attention has been focused on finding the relationship between fixed points and the minimum feedback arc set in interaction digraph of BNs [7]. In general, many real-world genetic regulatory networks are sparse. Thus, finding a minimum feedback arc set satisfying (5) is feasible in practice. After considering Problem 1, the controlled nodes can be easily determined, which are the ending vertices of edges in the minimal feedback arc sets in the interaction digraph of (1).

Assume that $\Omega_\varsigma = \{e_1^\varsigma, \ldots, e_{c_2}^\varsigma\} \in \{\Omega_1, \ldots, \Omega_\kappa\}$ ($\varsigma \in [1, \kappa]$) is one feasible minimal feedback arc set with cardinality c_2, which satisfies conditions (4) and (5). Then, we further assume

$$\bigcup_{i=1}^{c_2} O_+(e_i^\varsigma) \triangleq \{\omega_1, \ldots, \omega_{c_1}\}, \quad \omega_1, \ldots, \omega_{c_1} \in V, \quad (6)$$

which are the ending vertices of edges of the minimal feedback arc set $\Omega_\varsigma = \{e_1^\varsigma, \ldots, e_{c_2}^\varsigma\}$. Thus, under these assumptions, nodes x_i, $i \in \{\omega_1, \ldots, \omega_{c_1}\}$, should be controlled after considering Problem 1. Since some edges may share a same ending vertices, (6) implies that vertices $\omega_1, \ldots, \omega_{c_1}$ are all the possible ending vertices for the edges $e_1^\varsigma, \ldots, e_{c_2}^\varsigma$. Here, we assume that the edges $e_{l_1^1}^\varsigma, \ldots, e_{l_1^{\varepsilon_1}}^\varsigma \in \{e_1^\varsigma, \ldots, e_{c_2}^\varsigma\}$ ($l_1^1, \ldots, l_1^{\varepsilon_1} \in [1, c_2]$) share the same ending vertex ω_1, i.e. $\bigcup_{j=l_1^1}^{l_1^{\varepsilon_1}} O_+(e_j^\varsigma) \triangleq \{\omega_1\}$; ...; the edges $e_{l_{c_1}^1}^\varsigma, \ldots, e_{l_{c_1}^{\varepsilon_{c_1}}}^\varsigma \in \{e_1^\varsigma, \ldots, e_{c_2}^\varsigma\}$ ($l_{c_1}^1, \ldots, l_{c_1}^{\varepsilon_{c_1}} \in [1, c_2]$) share the same ending vertex ω_{c_1}, i.e. $\bigcup_{j=l_{c_1}^1}^{l_{c_1}^{\varepsilon_{c_1}}} O_+(e_j^\varsigma) \triangleq \{\omega_{c_1}\}$. In addition, $\bigcup_{j=l_1^1}^{l_1^{\varepsilon_1}} O_-(e_j^\varsigma) \triangleq \{\theta_1^1, \ldots, \theta_1^{\varepsilon_1}\}, \ldots, \bigcup_{j=l_{c_1}^1}^{l_{c_1}^{\varepsilon_{c_1}}} O_-(e_j^\varsigma) \triangleq \{\theta_{c_1}^1, \ldots, \theta_{c_1}^{\varepsilon_{c_1}}\}$. Here, assume that $\theta_1^1, \ldots, \theta_1^{\varepsilon_1}, \ldots, \theta_{c_1}^1, \ldots, \theta_{c_1}^{\varepsilon_{c_1}} \in V$, and $\theta_1^1 < \cdots < \theta_1^{\varepsilon_1}, \ldots, \theta_{c_1}^1 < \cdots < \theta_{c_1}^{\varepsilon_{c_1}}$.

After determining the controlled nodes based on the minimal feedback arc set given in Problem 1, the following problem is considered to further transform corresponding structure matrices of controlled nodes x_i, $i \in \{\omega_1, \ldots, \omega_{c_1}\}$ for designing NS-based distributed pinning control.

Problem 2 (Transformation of Structure Matrices of Pinning Nodes): Consider the nodes x_i, $i \in \{\omega_1, \ldots, \omega_{c_1}\}$ that should be controlled, find matrices $\tilde{A}_{\omega_1} \in \mathscr{L}_{2 \times 2^{|\mathscr{N}_{\omega_1}|}}, \ldots, \tilde{A}_{\omega_{c_1}} \in \mathscr{L}_{2 \times 2^{|\mathscr{N}_{\omega_{c_1}}|}}$, and $\hat{A}_{\omega_1} \in \mathscr{L}_{2 \times 2^{|\bar{\mathscr{N}}_{\omega_1}|}}, \ldots, \hat{A}_{\omega_{c_1}} \in \mathscr{L}_{2 \times 2^{|\bar{\mathscr{N}}_{\omega_{c_1}}|}}$, such that

$$\begin{cases} \tilde{A}_{\omega_1} &= \hat{A}_{\omega_1}(I_{2^{|\bar{\mathscr{N}}_{\omega_1}|}} \otimes \mathbf{1}_{2^{|\mathscr{N}_{\omega_1}^c|}}^\top), \\ &\cdots \\ \tilde{A}_{\omega_{c_1}} &= \hat{A}_{\omega_{c_1}}(I_{2^{|\bar{\mathscr{N}}_{\omega_{c_1}}|}} \otimes \mathbf{1}_{2^{|\mathscr{N}_{\omega_{c_1}}^c|}}^\top). \end{cases} \quad (7)$$

Here, we denote $\bar{\mathscr{N}}_{\omega_1} \triangleq \mathscr{N}_{\omega_1} \setminus \{\theta_1^1, \ldots, \theta_1^{\varepsilon_1}\}, \ldots, \bar{\mathscr{N}}_{\omega_{c_1}} \triangleq \mathscr{N}_{\omega_{c_1}} \setminus \{\theta_{c_1}^1, \ldots, \theta_{c_1}^{\varepsilon_{c_1}}\}$, and $\mathscr{N}_{\omega_1}^c \triangleq \{\theta_1^1, \ldots, \theta_1^{\varepsilon_1}\}, \ldots, \mathscr{N}_{\omega_{c_1}}^c \triangleq \{\theta_{c_1}^1, \ldots, \theta_{c_1}^{\varepsilon_{c_1}}\}$.

For example, consider matrix $\tilde{A}_{\omega_1}$ satisfying (7), i.e. $\tilde{A}_{\omega_1} = \hat{A}_{\omega_1}(I_{2^{|\bar{\mathscr{N}}_{\omega_1}|}} \otimes \mathbf{1}_{2^{|\mathscr{N}_{\omega_1}^c|}}^\top)$, then using STP method, suppose that the

algebraic form of node x_{ω_1} becomes $x_{\omega_1}(t+1) = \tilde{A}_{\omega_1} \ltimes_{j \in \bar{\mathcal{N}}_{\omega_j}}$ $x_j(t) \ltimes_{j \in \mathcal{N}^c_{\omega_j}} x_j(t) = \hat{A}_{\omega_1}(I_{2^{|\bar{\mathcal{N}}_{\omega_1}|}} \otimes \mathbf{1}^{\top}_{2^{|\mathcal{N}^c_{\omega_1}|}}) \ltimes_{j \in \bar{\mathcal{N}}_{\omega_j}} x_j(t) \ltimes_{j \in \mathcal{N}^c_{\omega_j}}$ $x_j(t)$, one can finally simplify as follows: $x_{\omega_1}(t+1) =$ $\hat{A}_{\omega_1} \ltimes_{j \in \bar{\mathcal{N}}_{\omega_1}} x_j(t)$. This implies that in the interaction digraph $G_{id} = (V, \tilde{E})$, the corresponding edges from nodes $x_{\theta_1^1}, \ldots, x_{\theta_1^{\varepsilon_1}}$ to node x_{ω_1} will be deleted. Hence, consider the algebraic forms of nodes $x_{\omega_1}, \ldots, x_{\omega_{\mathbf{c}_1}}$: $x_{\omega_1}(t+1) = A_{\omega_1} \ltimes_{j \in \mathcal{N}_{\omega_1}}$ $x_j(t), \ldots, x_{\omega_{\mathbf{c}_1}}(t+1) = A_{\omega_{\mathbf{c}_1}} \ltimes_{j \in \mathcal{N}_{\omega_{\mathbf{c}_1}}} x_j(t)$, after finding matrices $\tilde{A}_{\omega_1}, \ldots, \tilde{A}_{\omega_{\mathbf{c}_1}}$, and $\hat{A}_{\omega_1}, \ldots, \hat{A}_{\omega_{\mathbf{c}_1}}$ satisfying (7), if the structure matrices $A_{\omega_1}, \ldots, A_{\omega_{\mathbf{c}_1}}$ are transformed to matrices $\tilde{A}_{\omega_1}, \ldots, \tilde{A}_{\omega_{\mathbf{c}_1}}$, then the edges in the minimal feedback arc sets found in Problem 1 will be deleted in the digraph $G_{id} = (V, \tilde{E})$ which leads to an acyclic network structure. Note that there are many feasible matrices $\tilde{A}_{\omega_1}, \ldots, \tilde{A}_{\omega_{\mathbf{c}_1}}$, and $\hat{A}_{\omega_1}, \ldots, \hat{A}_{\omega_{\mathbf{c}_1}}$ satisfying (7), one can firstly choose matrices $\hat{A}_{\omega_1}, \ldots, \hat{A}_{\omega_{\mathbf{c}_1}}$ in order to determine $\tilde{A}_{\omega_1}, \ldots, \tilde{A}_{\omega_{\mathbf{c}_1}}$ satisfying (7). Since matrices $\hat{A}_{\omega_1}, \ldots, \hat{A}_{\omega_{\mathbf{c}_1}}$ are logical matrices with finite dimensions, there only exist finite possible cases for $\hat{A}_{\omega_1}, \ldots, \hat{A}_{\omega_{\mathbf{c}_1}}$.

Using the property of swap matrix $W_{[m,n]}$, that is $W_{[m,n]}\sigma_1 \ltimes \sigma_2 = \sigma_2 \ltimes \sigma_1$, for $x_{\omega_j}(t+1) = A_{\omega_j} \ltimes_{j \in \mathcal{N}_{\omega_j}} x_j(t), j \in [1, \mathbf{c}_1]$, one can swap the positions of neighbors as follows:

$$\begin{aligned} x_{\omega_j}(t+1) = \quad & A_{\omega_j} \mathbf{W}_j \ltimes_{j \in \bar{\mathcal{N}}_{\omega_j}} x_j(t) \ltimes_{j \in \mathcal{N}^c_{\omega_j}} x_j(t), \\ \triangleq \quad & \bar{A}_{\omega_j} \ltimes_{j \in \bar{\mathcal{N}}_{\omega_j}} x_j(t) \ltimes_{j \in \mathcal{N}^c_{\omega_j}} x_j(t), \end{aligned} \tag{8}$$

where $\mathbf{W}_j \triangleq [\ltimes_{i=1}^{\omega_j} W_{[2,2^{|\theta_j^i|\mathcal{N}_{\omega_j}|}]} W_{[2^{|\mathcal{N}^c_{\omega_j}|}, 2^{|\mathcal{N}_{\omega_j}|}]}] \otimes I_{2^{|\mathcal{N}_{\omega_j}| - |\mathcal{N}^c_{\omega_j}|}}$ are invertible matrices, and $\bar{A}_{\omega_j} \triangleq A_{\omega_j} \mathbf{W}_j$.

Then, based on Problems 1 and 2, one can design an NS-based distributed pinning control to guarantee global stabilization on nodes $x_j, j \in \{\omega_1, \ldots, \omega_{\mathbf{c}_1}\}$, in the following form:

$$\begin{cases} x_j(t+1) = & u_j(t) \oplus_j f_j([x_{j_1}(t)]_{j_1 \in \mathcal{N}_j}), j \in \{\omega_1, \ldots, \omega_{\mathbf{c}_1}\}, \\ u_j(t) = & g_j([x_{j_1}(t)]_{j_1 \in \mathcal{N}_j}), \qquad j \in \{\omega_1, \ldots, \omega_{\mathbf{c}_1}\}, \\ x_i(t+1) = & f_i([x_{j_2}(t)]_{j_2 \in \mathcal{N}_i}), \quad i \in [1, n] \backslash \{\omega_1, \ldots, \omega_{\mathbf{c}_1}\}. \end{cases} \tag{9}$$

Here, $u_j \in \mathscr{D}, j \in \{\omega_1, \ldots, \omega_{\mathbf{c}_1}\}$, are NS-based distributed pinning controllers for nodes $x_j, j \in \{\omega_1, \ldots, \omega_{\mathbf{c}_1}\}$. $\oplus_j : \mathscr{D}^2 \to \mathscr{D}, j \in \{\omega_1, \ldots, \omega_{\mathbf{c}_1}\}$, are binary logical functions to be determined in the following sequel. Functions $g_j : \mathscr{D}^{|\mathcal{N}_j|} \to \mathscr{D}, j \in \{\omega_1, \ldots, \omega_{\mathbf{c}_1}\}$, are Boolean functions of controllers u_j depending on the corresponding neighbors of nodes x_j $(j \in \{\omega_1, \ldots, \omega_{\mathbf{c}_1}\})$, which will be determined later. Since any logical function can be equivalently expressed by means of suitable structure matrix, let $M_{\oplus_j} \in \mathscr{L}_{2 \times 4}$ and $K_j \in \mathscr{L}_{2 \times 2^{|\mathcal{N}_j|}}$, $j \in \{\omega_1, \ldots, \omega_{\mathbf{c}_1}\}$, denote the structure matrices of functions $\oplus_j : \mathscr{D}^2 \to \mathscr{D}$ and $g_j : \mathscr{D}^{|\mathcal{N}_j|} \to \mathscr{D}$. Thus, in the following, one needs to determine the structure matrices $M_{\oplus_j}$ and K_j, to determine the logical functions $\oplus_j : \mathscr{D}^2 \to \mathscr{D}$ and $g_j : \mathscr{D}^{|\mathcal{N}_j|} \to \mathscr{D}$.

Let $\mathbf{K}_j \triangleq K_j \mathbf{W}_j \in \mathscr{L}_{2 \times 2^{|\mathcal{N}_j|}}, j \in \{\omega_1, \ldots, \omega_{\mathbf{c}_1}\}$. For nodes $x_j, j \in \{\omega_1, \ldots, \omega_{\mathbf{c}_1}\}$, using STP, Eq. (9) can be expressed as follows:

$$\begin{cases} x_j(t+1) = M_{\oplus_j} u_j(t) \bar{A}_j \ltimes_{i \in \bar{\mathcal{N}}_j} x_i(t) \ltimes_{i \in \mathcal{N}^c_j} x_i(t), \\ \qquad = M_{\oplus_j} \mathbf{K}_j (I_{2^{|\mathcal{N}_j|}} \otimes \bar{A}_j) \Phi_{2^{|\mathcal{N}_j|}} \ltimes_{i \in \bar{\mathcal{N}}_j} x_i(t) \ltimes_{i \in \mathcal{N}^c_j} x_i(t), \\ u_j(t) = \mathbf{K}_j \ltimes_{i \in \bar{\mathcal{N}}_j} x_i(t) \ltimes_{i \in \mathcal{N}^c_j} x_i(t). \end{cases} \tag{10}$$

According to Problems 1 and 2, suppose that for node x_j, $j \in \{\omega_1, \ldots, \omega_{\mathbf{c}_1}\}$, system $x_j(t+1) =$ $M_{\oplus_j} \mathbf{K}_j (I_{2^{|\mathcal{N}_j|}} \otimes \bar{A}_j) \Phi_{2^{|\mathcal{N}_j|}} \ltimes_{i \in \bar{\mathcal{N}}_j} x_i(t) \ltimes_{i \in \mathcal{N}^c_j} x_i(t)$ becomes $x_j(t+1) = \hat{A}_j (I_{2^{|\bar{\mathcal{N}}_j|}} \otimes \mathbf{1}^{\top}_{2^{|\mathcal{N}^c_j|}}) \ltimes_{i \in \bar{\mathcal{N}}_j} x_i(t) \ltimes_{i \in \mathcal{N}^c_j} x_i(t)$, then one can obtain $x_j(t+1) = \hat{A}_j \ltimes_{i \in \bar{\mathcal{N}}_j} x_i(t)$. Thus, according to $x_j(t+1) = \hat{A}_j \ltimes_{i \in \bar{\mathcal{N}}_j} x_i(t)$ $(j \in \{\omega_1, \ldots, \omega_{\mathbf{c}_1}\})$, in the interaction digraph $G_{id} = (V, \tilde{E})$, the edges from nodes $x_{v_1^1}, \cdots, x_{v_1^{\varepsilon_1}}$ to node x_{ω_1}, $\cdots$, and the edges from nodes $x_{v_{\mathbf{c}_1}^1}, \cdots, x_{v_{\mathbf{c}_1}^{\varepsilon_{\mathbf{c}_1}}}$ to node $x_{\omega_{\mathbf{c}_1}}$ will be deleted. This implies that if there exist matrices $M_{\oplus_j} \in \mathscr{L}_{2 \times 4}$ and $K_j \in \mathscr{L}_{2 \times 2^{|\mathcal{N}_j|}}, j \in \{\omega_1, \ldots, \omega_{\mathbf{c}_1}\}$ satisfying $\tilde{A}_{\omega_j} = M_{\oplus_j} \mathbf{K}_{\omega_j} (I_{2^{|\mathcal{N}_{\omega_j}|}} \otimes \bar{A}_{\omega_j}) \Phi_{2^{|\mathcal{N}_{\omega_j}|}}$, then the interaction digraph $G_{id} = (V, \tilde{E})$ will become acyclic.

In order to design NS-based distributed pinning controllers, one needs to solve matrices $M_{\oplus_j} \in \mathscr{L}_{2 \times 4}$ and $K_j \in \mathscr{L}_{2 \times 2^{|\mathcal{N}_j|}}$, $j \in \{\omega_1, \ldots, \omega_{\mathbf{c}_1}\}$ satisfying the equations below,

$$\begin{cases} \tilde{A}_{\omega_1} & = M_{\oplus_{\omega_1}} \mathbf{K}_{\omega_1} (I_{2^{|\mathcal{N}_{\omega_1}|}} \otimes \bar{A}_{\omega_1}) \Phi_{2^{|\mathcal{N}_{\omega_1}|}}, \\ & \cdots \\ \tilde{A}_{\omega_{\mathbf{c}_1}} & = M_{\oplus_{\omega_{\mathbf{c}_1}}} \mathbf{K}_{\omega_{\mathbf{c}_1}} (I_{2^{|\mathcal{N}_{\omega_{\mathbf{c}_1}}|}} \otimes \bar{A}_{\omega_{\mathbf{c}_1}}) \Phi_{2^{|\mathcal{N}_{\omega_{\mathbf{c}_1}}|}}. \end{cases} \tag{11}$$

Thus, the solvability of (11) is vital for designing NS-based distributed pinning controllers, which is guaranteed by the following theorem.

Theorem 3.1: Given matrices $\tilde{A}_{\omega_j}$ and $\bar{A}_{\omega_j}, j \in [1, \mathbf{c}_1]$, one can obtain that Eq. (11) is always solvable.

Proof. Here, for simplicity, we only prove the solvability of equation $\tilde{A}_{\omega_1} = M_{\oplus_{\omega_1}} \mathbf{K}_{\omega_1} (I_{2^{|\mathcal{N}_{\omega_1}|}} \otimes \bar{A}_{\omega_1}) \Phi_{2^{|\mathcal{N}_{\omega_1}|}}$, the rest of (11) can be similarly derived. Note that $\tilde{A}_{\omega_1} = M_{\oplus_{\omega_1}} \mathbf{K}_{\omega_1} (I_{2^{|\mathcal{N}_{\omega_1}|}} \otimes \bar{A}_{\omega_1}) \Phi_{2^{|\mathcal{N}_{\omega_1}|}} = M_{\oplus_{\omega_1}} (\mathbf{K}_{\omega_1} \otimes \bar{A}_{\omega_1}) \Phi_{2^{|\mathcal{N}_{\omega_1}|}}$. For simplicity, here, denote $|\mathcal{N}_{\omega_1}| = n_1$, and assume that matrix $M_{\oplus_{\omega_1}} = \begin{bmatrix} \alpha_1 & \alpha_2 & \alpha_3 & \alpha_4 \\ 1-\alpha_1 & 1-\alpha_2 & 1-\alpha_3 & 1-\alpha_4 \end{bmatrix} \in \mathscr{L}_{2 \times 4}$, matrix $\mathbf{K}_{\omega_1} = \begin{bmatrix} \beta_1 & \beta_2 & \cdots & \beta_{2^{n_1}} \\ 1-\beta_1 & 1-\beta_2 & \cdots & 1-\beta_{2^{n_1}} \end{bmatrix} \in \mathscr{L}_{2 \times 2^{n_1}}$, matrix $\bar{A}_{\omega_1} = \begin{bmatrix} y_1 & y_2 & \cdots & y_{2^{n_1}} \\ 1-y_1 & 1-y_2 & \cdots & 1-y_{2^{n_1}} \end{bmatrix} \in \mathscr{L}_{2 \times 2^{n_1}}$, and matrix $\tilde{A}_{\omega_1} = \begin{bmatrix} a_1 & a_2 & \cdots & a_{2^{n_1}} \\ 1-a_1 & 1-a_2 & \cdots & 1-a_{2^{n_1}} \end{bmatrix} \in \mathscr{L}_{2 \times 2^{n_1}}$, where $\alpha_1, \alpha_2, \alpha_3, \alpha_4 \in \mathscr{D}, \beta_j, y_j, a_j \in \mathscr{D}, j \in [1, 2^{n_1}]$. Then, one can obtain the following equation,

$$\begin{aligned} \tilde{A}_{\omega_1} &= M_{\oplus_{\omega_1}} \mathbf{K}_{\omega_1} (I_{2^{|\mathcal{N}_{\omega_1}|}} \otimes \bar{A}_{\omega_1}) \Phi_{2^{|\mathcal{N}_{\omega_1}|}} \\ &= \begin{bmatrix} \alpha_1 & \alpha_2 & \alpha_3 & \alpha_4 \\ 1-\alpha_1 & 1-\alpha_2 & 1-\alpha_3 & 1-\alpha_4 \end{bmatrix} \times \\ & \begin{bmatrix} \beta_1 y_1 & \cdots & \beta_{2^{n_1}} y_{2^{n_1}} \\ \beta_1(1-y_1) & \cdots & \beta_{2^{n_1}}(1-y_{2^{n_1}}) \\ (1-\beta_1)y_1 & \cdots & (1-\beta_{2^{n_1}})y_{2^{n_1}} \\ (1-\beta_1)(1-y_1) & \cdots & (1-\beta_{2^{n_1}})(1-y_{2^{n_1}}) \end{bmatrix} \Phi_{2^{n_1}} \\ &= \begin{bmatrix} a_1 & a_2 & \cdots & a_{2^{n_1}} \\ 1-a_1 & 1-a_2 & \cdots & 1-a_{2^{n_1}} \end{bmatrix}. \end{aligned} \tag{12}$$

By simplifying (12), the solvability of equation $\tilde{A}_{\omega_1} = M_{\oplus_{\omega_1}} \mathbf{K}_{\omega_1} (I_{2^{|\mathcal{N}_{\omega_1}|}} \otimes \bar{A}_{\omega_1}) \Phi_{2^{|\mathcal{N}_{\omega_1}|}}$ is converted to the following equations:

$$\begin{aligned} \alpha_1 \beta_i y_i + \alpha_2 \beta_i(1-y_i) + \alpha_3(1-\beta_i)y_i + \\ \alpha_4(1-\beta_i)(1-y_i) = a_i, \quad i \in [1, 2^{n_1}]. \end{aligned} \tag{13}$$

Since matrices $\tilde{A}_{\omega_1}$ and $\bar{A}_{\omega_1}$ are known, we only need to consider the following four cases to determine the values of α_i and β_i according to different values of y_i and a_i.

Consider (13), as for case 1: $a_i = 1, y_i = 1$, then (13) becomes $\alpha_1 \beta_i + \alpha_3 (1 - \beta_i) = 1$. Then, $\alpha_1 \beta_i + \alpha_3 (1 - \beta_i) = 1$ holds under solutions $\beta_i = 1$ and $\alpha_1 = 1$ for any $\alpha_2, \alpha_3, \alpha_4 \in \{1, 0\}$.

As for case 2: $a_i = 1, y_i = 0$, (13) holds under solution $\beta_i = 0, \alpha_4 = 1$ for any $\alpha_2, \alpha_3, \alpha_1 \in \{1, 0\}$.

As for case 3: $a_i = 0, y_i = 1$, (13) holds under solution $\beta_i = 0, \alpha_3 = 0$ for any $\alpha_1, \alpha_2, \alpha_4 \in \{1, 0\}$.

As for case 4: $a_i = 0, y_i = 0$, (13) holds under solution $\beta_i = 1, \alpha_2 = 0$ for any $\alpha_1, \alpha_3, \alpha_4 \in \{1, 0\}$. Thus, according to the above analysis, equation $\tilde{A}_{\omega_1} = M_{\oplus_{\omega_1}} \mathbf{K}_{\omega_1} (I_{2^{|\mathcal{N}_{\omega_1}|}} \otimes \bar{A}_{\omega_1}) \Phi_{2^{|\mathcal{N}_{\omega_1}|}}$ is always solvable. Then, similarly, one can also conclude that Eq. (11) is solvable, which completes the proof. ∎

Based on the solvability of (11), global stabilization of BN under an NS-based distributed pinning control in the form of Eq. (9) will be guaranteed by the following theorem.

Theorem 3.2: By solving matrices $M_{\oplus_j} \in \mathcal{L}_{2 \times 4}$ and $K_j \in \mathcal{L}_{2 \times 2^{|\mathcal{N}_j|}}$, $j \in \{\omega_1, \ldots, \omega_{\mathbf{c}_1}\}$ from Eq. (11), BN under an NS-based distributed pinning control in the form of Eq. (9) will be globally stabilized.

Proof. If BN (1) is under the NS-based distributed pinning control in the form of (9), then one can obtain its algebraic form for the controlled nodes x_j, $j \in \{\omega_1, \ldots, \omega_{\mathbf{c}_1}\}$,

$$
\begin{aligned}
x_j(t+1) &= M_{\oplus_j} u_j(t) \bar{A}_j \ltimes_{i \in \bar{\mathcal{N}}_j} x_i(t) \ltimes_{i \in \mathcal{N}_j^c} x_i(t), \\
&= M_{\oplus_j} \mathbf{K}_j (I_{2^{|\mathcal{N}_j|}} \otimes \bar{A}_j) \Phi_{2^{|\mathcal{N}_j|}} \ltimes_{i \in \bar{\mathcal{N}}_j} x_i(t) \ltimes_{i \in \mathcal{N}_j^c} x_i(t).
\end{aligned}
$$

Since matrices $M_{\oplus_j} \in \mathcal{L}_{2 \times 4}$ and $K_j \in \mathcal{L}_{2 \times 2^{|\mathcal{N}_j|}}$, $j \in \{\omega_1, \ldots, \omega_{\mathbf{c}_1}\}$ can be obtained by solving Eq. (11), one has

$$
x_j(t+1) = \tilde{A}_{\omega_j} \ltimes_{i \in \bar{\mathcal{N}}_j} x_i(t) \ltimes_{i \in \mathcal{N}_j^c} x_i(t).
$$

By considering Problem 2, one can find matrices $\tilde{A}_{\omega_j}$, $j \in \{1, \ldots, \mathbf{c}_1\}$ satisfying Eq. (7). Further, one can obtain that

$$
\begin{aligned}
x_{\omega_j}(t+1) &= \hat{A}_{\omega_j} (I_{2^{|\mathcal{N}_{\omega_j}|}} \otimes \mathbf{1}_{2^{|\mathcal{N}_{\omega_j}^c|}}^\top) \ltimes_{i \in \bar{\mathcal{N}}_j} x_i(t) \ltimes_{i \in \mathcal{N}_j^c} x_i(t) \\
&= \hat{A}_{\omega_j} \ltimes_{i \in \bar{\mathcal{N}}_j} x_i(t), j \in \{\omega_1, \ldots, \omega_{\mathbf{c}_1}\}.
\end{aligned}
$$

This implies that in the interaction digraph $G_{id} = (V, E)$, the NS-based distributed pinning controllers (9) will delete the edges from nodes $x_{\theta_1^1}, \cdots, x_{\theta_1^{\varepsilon_1}}$ to node $x_{\omega_1}, \cdots$, and the edges from nodes $x_{\theta_{\mathbf{c}_1}^1}, \cdots, x_{\theta_{\mathbf{c}_1}^{\varepsilon_{\mathbf{c}_1}}}$ to node $x_{\omega_{\mathbf{c}_1}}$. According to Problem 1, these deleted edges lead the interaction digraph $G_{id} = (V, E)$ to be acyclic. Then, based on Lemma 2.2, system (9) is globally stabilized, which completes the proof. ∎

After considering Step 1, an NS-based distributed pinning control in the form of (9) is designed in order to guarantee global stabilization, but the designed pinning control (9) may not guarantee global stabilization to the given fixed point $\delta_{2^n}^{\gamma}$. Note that fixed points of BNs is used to represent stable configuration of cell types of genetic regulatory networks, such as cell death or unregulated growth. Thus, it is a significant issue to further design pinning control method to achieve global stabilization to a given fixed point, which will be studied in the following step. Under the NS-based distributed pinning control design in the form of (9), with corresponding

algebraic form (10), one has the reduced system of (2):

$$
\begin{cases}
x_s(t+1) = \hat{A}_s \ltimes_{i \in \bar{\mathcal{N}}_s} x_i(t), & s \in \{\omega_1, \ldots, \omega_{\mathbf{c}_1}\}, \\
x_t(t+1) = A_t \ltimes_{i \in \mathcal{N}_t} x_i(t), & t \in [1, n] \setminus \{\omega_1, \ldots, \omega_{\mathbf{c}_1}\}.
\end{cases}
\tag{14}
$$

As it is seen from (14), we have imposed state feedback pinning controllers on nodes x_j, $j \in \{\omega_1, \ldots, \omega_{\mathbf{c}_1}\}$, while there is no controller imposed on nodes x_i, $i \in [1, n] \setminus \{\omega_1, \ldots, \omega_{\mathbf{c}_1}\}$. Thus, the structure matrices for x_i, $i \in [1, n] \setminus \{\omega_1, \ldots, \omega_{\mathbf{c}_1}\}$, remain unchanged. However, in order to avoid confusion, we still denote $\hat{A}_i = A_i$, $\bar{\mathcal{N}}_i = \mathcal{N}_i$, $i \in [1, n] \setminus \{\omega_1, \ldots, \omega_{\mathbf{c}_1}\}$, for convenience of the following step.

B. Step 2: achieve global stabilization to a given state $\delta_{2^n}^{\gamma}$

In the following, one needs to further design a state feedback pinning control to guarantee that (14) will achieve global stabilization to the given fixed point $\delta_{2^n}^{\gamma}$. In order to further achieve global stabilization to the given fixed point $\delta_{2^n}^{\gamma}$, the following problem is considered to further determine the controlled nodes that are needed to achieve global stabilization to a given fixed point $\delta_{2^n}^{\gamma}$, the corresponding structure matrices of which will be changed under the design of state feedback pinning control.

Problem 3 (Determination of Pinning Nodes and Transformation of Structure Matrices): Let $\delta_{2^n}^{\gamma} = \ltimes_{i=1}^n \delta_2^{\gamma_i}$. Find matrices $\check{A}_1 \in \mathcal{L}_{2 \times 2^{|\bar{\mathcal{N}}_1|}}, \ldots, \check{A}_n \in \mathcal{L}_{2 \times 2^{|\bar{\mathcal{N}}_n|}}$ and binary variables $\delta_1, \ldots, \delta_n \in \mathcal{D}$ that minimize the cost function

$$
\mathbf{c}_3 \triangleq \sum_{i=1}^n \delta_i
\tag{15}
$$

subject to the following conditions:

$$
\delta_2^{\gamma_i} = [(1 - \delta_i) \times \hat{A}_i + \delta_i \times \check{A}_i] \ltimes_{j \in \bar{\mathcal{N}}_i} \delta_2^{\gamma_j}, \quad i \in [1, n].
\tag{16}
$$

Problem 3 can be rewritten as an integer linear programming (ILP) problem, and can be solved by a suitable solver like Yices SMT Solver [33]. Here, (16) is given to search structure matrices among $\hat{A}_j$ ($j \in [1, n]$) that does not satisfy the condition of the given fixed point $\delta_{2^n}^{\gamma}$ for (14). As one can see from (16), if $\delta_1 = 1$, $\delta_2^{\gamma_1} = [(1 - \delta_1) \hat{A}_1 + \delta_1 \check{A}_1] \ltimes_{j \in \bar{\mathcal{N}}_1} \delta_2^{\gamma_j}$ reduces to $\delta_2^{\gamma_1} = \check{A}_1 \ltimes_{j \in \bar{\mathcal{N}}_1} \delta_2^{\gamma_j}$, this implies that node x_1 should be controlled and its structure matrix $\hat{A}_1$ should be changed to another structure matrix $\check{A}_1$ satisfying the condition of the given fixed point $\delta_{2^n}^{\gamma}$ for (14). If $\delta_1 = 0$, $\delta_2^{\gamma_1} = [(1 - \delta_1) \hat{A}_1 + \delta_1 \check{A}_1] \ltimes_{j \in \bar{\mathcal{N}}_1} \delta_2^{\gamma_j}$ reduces to $\delta_2^{\gamma_1} = \hat{A}_1 \ltimes_{j \in \bar{\mathcal{N}}_1} \delta_2^{\gamma_j}$, this implies that there is no need to transform the structure matrix $\hat{A}_1$ for node x_1. Thus, $\sum_{i=1}^n \delta_i$ represents the number of nodes requiring to be further controlled, and by minimizing the cost function $\mathbf{c}_3 = \sum_{i=1}^n \delta_i$, one can further determine the minimum number of controlled nodes. In addition, the determination of controlled nodes is given by searching non-zero variables among variables $\delta_1, \ldots, \delta_n \in \mathcal{D}$.

A simple Example 3.1 is presented in Appendix A to illustrate problem 1 and how to determine pinning nodes for further achieving global stabilization to a given state $\delta_{2^n}^{\gamma}$.

By considering Problem 3, one can find matrices $\check{A}_1 \in \mathcal{L}_{2 \times 2^{|\bar{\mathcal{N}}_1|}}, \ldots, \check{A}_n \in \mathcal{L}_{2 \times 2^{|\bar{\mathcal{N}}_n|}}$ and binary variables $\delta_1, \ldots, \delta_n \in \mathcal{D}$, such that the cost function $\mathbf{c}_3$ is minimized. Here, it is assumed that under the solution $\delta_{\tau_1} = \delta_{\tau_2} = \cdots = \delta_{\tau_l} = 1$,

$\delta_j = 0$ $(j \in \{1,\ldots,n\}\backslash\{\tau_1,\ldots,\tau_l\})$, the cost function $\mathbf{c}_3$ is minimum, that is $\mathbf{c}_3 = l$, with the corresponding matrices $\check{A}_p$, $p \in \{\tau_1,\ldots,\tau_l\}$. Since the determination of controlled nodes is given by non-zero variables among $\delta_1,\ldots,\delta_n \in \mathscr{D}$, then nodes x_p, $p \in \{\tau_1,\ldots,\tau_l\}$ will be further controlled in order to achieve global stabilization to the given fixed point $\delta_{2^n}^\gamma$.

Then, further design an NS-based distributed pinning control to achieve global stabilization to a given fixed point $\delta_{2^n}^\gamma$. The controllers are imposed on nodes x_p, $p \in \{\tau_1,\ldots,\tau_l\}$,

$$\begin{cases} x_p(t+1) &= \hat{u}_p(t)\hat{\oplus}_p \hat{f}_p([x_j(t)]_{j\in\bar{\mathscr{N}}_p}), \\ \hat{u}_p(t) &= \hat{g}_p([x_j(t)]_{j\in\bar{\mathscr{N}}_p}), \end{cases} \quad (17)$$

while the rest of the nodes x_j, $j \in \{1,\ldots,n\}\backslash\{\tau_1,\ldots,\tau_l\}$, remain unchanged. In addition, the structure matrices of logical functions $\hat{f}_p([x_j(t)]_{j\in\bar{\mathscr{N}}_p})$, $p \in \{\tau_1,\ldots,\tau_l\}$, are $\hat{A}_p$, which are found in Problem 2.

Suppose that the structure matrices for logical functions $\hat{g}_p$, $p \in \{\tau_1,\ldots,\tau_l\}$, are denoted by $\hat{K}_p \in \mathscr{L}_{2\times 2^{|\bar{\mathscr{N}}_p|}}$, and the structure matrices of function $\hat{\oplus}_p$, $p \in \{\tau_1,\ldots,\tau_l\}$, are denoted by $\hat{M}_{\hat{\oplus}_p} \in \mathscr{L}_{2\times 4}$. Then, for $p \in \{\tau_1,\ldots,\tau_l\}$, one can firstly obtain the algebraic form of (17) as follows:

$$\begin{cases} x_p(t+1) = \hat{M}_{\hat{\oplus}_p}\hat{u}_p(t)\hat{A}_p \ltimes_{j\in\bar{\mathscr{N}}_p} x_j(t), \\ \qquad = \hat{M}_{\hat{\oplus}_p}\hat{K}_p(I_{2^{|\bar{\mathscr{N}}_p|}}\otimes\hat{A}_p)\Phi_{2^{|\bar{\mathscr{N}}_p|}} \ltimes_{j\in\bar{\mathscr{N}}_p} x_j(t), \quad (18) \\ \hat{u}_p(t) = \hat{K}_p \ltimes_{j\in\bar{\mathscr{N}}_p} x_j(t). \end{cases}$$

Thus, in order to determine matrices $\hat{K}_p \in \mathscr{L}_{2\times 2^{|\bar{\mathscr{N}}_p|}}$, $\hat{M}_{\hat{\oplus}_p} \in \mathscr{L}_{2\times 4}$, $p \in \{\tau_1,\ldots,\tau_l\}$, one needs to solve the following equations for $p \in \{\tau_1,\ldots,\tau_l\}$:

$$\check{A}_p = \hat{M}_{\hat{\oplus}_p}\hat{K}_p(I_{2^{|\bar{\mathscr{N}}_p|}}\otimes\hat{A}_p)\Phi_{2^{|\bar{\mathscr{N}}_p|}}. \quad (19)$$

It is noted that Eq. (19) is also solvable compared with (11), which can be similarly proved using Theorem 3.1, thus it is implementable to design state feedback controllers to achieve global stabilization to the prescribed fixed point $\delta_{2^n}^\gamma$.

In summary, for simplicity, here, we denote four sets $\mathscr{U}_+ = \{\omega_1,\ldots,\omega_{\mathbf{c}_1}\}\bigcup\{\tau_1,\ldots,\tau_l\}$, $\mathscr{U}_- = \{\omega_1,\ldots,\omega_{\mathbf{c}_1}\}\bigcap\{\tau_1,\ldots,\tau_l\}$, $\mathscr{U}_\omega = \{\omega_1,\ldots,\omega_{\mathbf{c}_1}\}\backslash\mathscr{U}_-$, $\mathscr{U}_\tau = \{\tau_1,\ldots,\tau_l\}\backslash\mathscr{U}_-$. Finally, after considering Steps 1 and 2, an NS-based distributed pinning control for global stabilization to a given fixed point $\delta_{2^n}^\gamma$ is designed as follows:

$$\begin{cases} x_{j_1}(t+1) &= \hat{u}_{j_1}\hat{\oplus}_{j_1}\left[u_{j_1}\oplus_{j_1}f_{j_1}([x_i(t)]_{i\in\mathscr{N}_{j_1}})\right], j_1 \in \mathscr{U}_-, \\ x_{j_2}(t+1) &= \left[u_{j_1}\oplus_{j_2}f_{j_2}([x_i(t)]_{i\in\mathscr{N}_{j_2}})\right], \qquad j_2 \in \mathscr{U}_\omega, \\ x_{j_3}(t+1) &= \left[\hat{u}_{j_3}\hat{\oplus}_{j_3}f_{j_3}([x_i(t)]_{i\in\mathscr{N}_{j_3}})\right], \qquad j_3 \in \mathscr{U}_\tau, \\ x_{j_4}(t+1) &= f_{j_4}([x_i(t)]_{i\in\mathscr{N}_{j_4}}), \qquad j_4 \in [1,n]\backslash\mathscr{U}_+. \end{cases} \quad (20)$$

Here, u_j and $\oplus_j$, $j \in \{\omega_1,\ldots,\omega_{\mathbf{c}_1}\}$, are controllers obtained in Step 1, while $\hat{u}_j$ and $\hat{\oplus}_j$, $j \in \{\tau_1,\ldots,\tau_l\}$, are controllers obtained in Step 2. Hence, this completes the main procedures of designing an NS-based distributed pinning control for global stabilization to a given fixed point $\delta_{2^n}^\gamma$.

Remark 3.2: During the past few decades, many fundamental results using the state transition matrix L have been established [9]–[12]. Unfortunately, there still exists several disadvantages concerning the state transition matrix L. One is

that using system $x(t+1) = Lx(t)$, the information of network structure is missing, such as nodes' connection, while matrix L only reflects the state transition digraph. Another one is that the dimension of L grows drastically with the size of network, which implies that the existing methods are difficult to be implemented for large-scale BNs. In this paper, a new approach for designing an NS-based distributed pinning control approach is proposed based on network structure of BNs without using the state transition matrix L. This paper aims to solve the problems of computational complexity (reduced from $O(2^n \times 2^n)$ to $O(2 \times 2^K)$, where n is the number of nodes and K is the largest number of in-neighbors of nodes).

IV. APPLICATIONS TO T-LGL SURVIVAL SIGNALING AND T-CELL RECEPTOR SIGNALING NETWORKS

In this section, several biological networks are presented to demonstrate the validity of the obtained results, the first example is a network with 6 nodes, the second one has 90 nodes, and the last one can be found in Appendix.

*Example 4.1 (**T-LGL Survival Signaling Network**):* Consider a reduced network in the T-LGL survival signaling network, which consists of six nodes, that is S1P, FLIP, Fas, Ceramide, DISC, Apoptosis [34]. Here, x_1,x_2,x_3,x_4,x_5,x_6 are used to represent these six nodes, respectively. The logical dynamics are given as follows:

$$\begin{cases} x_1(*) = \neg(x_4 \vee x_6), \qquad x_2(*) = \neg(x_5 \vee x_6), \\ x_3(*) = \neg(x_1 \vee x_6), \qquad x_4(*) = x_3 \vee \neg(x_1 \vee x_6), \quad (21) \\ x_5(*) = [x_4 \vee (x_3 \wedge \neg x_2)]\wedge\neg x_6, \quad x_6(*) = x_5 \vee x_6. \end{cases}$$

The symbol $(*)$ indicates the state of next step of the marked node to reduce space. For each node x_i, $i \in [1,6]$, one has its corresponding in-neighbors, that is $\mathscr{N}_1 = \{4,6\}$, $\mathscr{N}_2 = \{5,6\}$, $\mathscr{N}_3 = \{1,6\}$, $\mathscr{N}_4 = \{1,3,6\}$, $\mathscr{N}_5 = \{2,3,4,6\}$ and $\mathscr{N}_6 = \{5,6\}$. By simple calculations, one can conclude that system (21) is not globally stable to state δ_{64}^{31}. Now, we consider how to design an efficient NS-based distributed pinning control to achieve global stabilization to state δ_{64}^{31}.

Thus, by considering Steps 1 and 2, one can finally design the following NS-based distributed pinning controllers $u_1,u_5,u_6,\hat{u}_1$, which are imposed on nodes x_1,x_5 and x_6,

$$\begin{cases} x_1(*) = \hat{u}_1 \wedge \neg[u_1 \leftrightarrow \neg(x_4 \vee x_6)], \\ x_2(*) = \neg(x_5 \vee x_6), \qquad x_3(*) = \neg(x_1 \vee x_6), \\ x_4(*) = x_3 \vee \neg(x_1 \vee x_6), \qquad x_6(*) = u_6 \vee x_5 \vee x_6, \quad (22) \\ x_5(*) = u_5 \leftrightarrow \{[x_4 \vee (x_3 \wedge \neg x_2)]\wedge\neg x_6\}. \end{cases}$$

The corresponding NS-based distributed pinning controllers $u_1,u_5,u_6,\hat{u}_1$ are designed as follows:

$$\begin{cases} u_1 = x_4 \to x_6, \quad \hat{u}_1 = x_6, \quad u_6 = \neg x_5 \wedge \neg x_6, \\ u_5 = [x_2 \wedge (x_3 \vee x_6)] \vee \neg x_2. \end{cases} \quad (23)$$

Due to the page limitation, the detailed derivations for the above pinning control (22) and (23) are omitted here, which can be obtained using the proposed method in this paper. Thus, by controlling nodes x_1,x_5,x_6 and four corresponding NS-based distributed pinning controllers $u_1,u_5,u_6,\hat{u}_1$ in the form of (22) and (23), system (22) is globally stabilized to state δ_{64}^{31}. The state transition graphs for both (21) and (22) under pinning control (23), are shown in Fig. 2(a) and Fig. 2(b). Using the
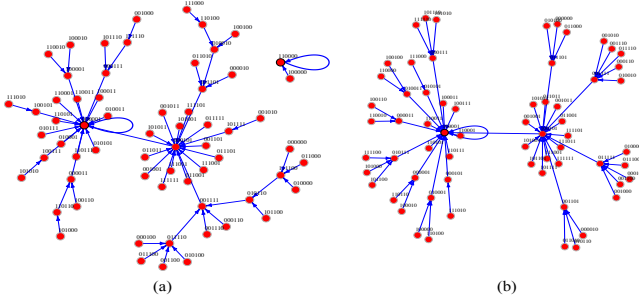
Fig. 2.   (a) State transition digraph of (21) with two fixed points $((0,0,0,0,0,1)$ and $(1,1,0,0,0,0))$; (b) State transition graph of (22) under pinning control strategies (23) with a unique fixed point $(1,0,0,0,0,1)$. Each node represents a state of the network, and each arrow is a state transition.
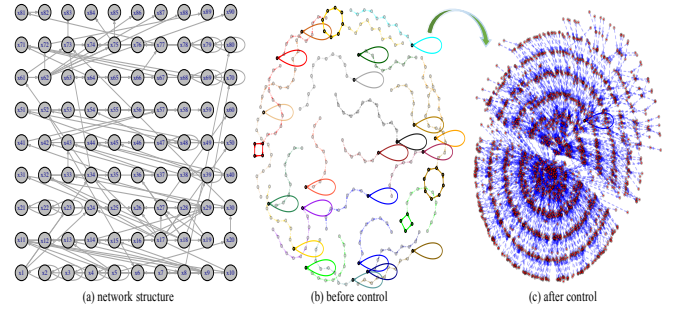

Fig. 3.   (a) Network structure of T-Cell receptor signaling Boolean model [35]; (b) Part of state transition digraph randomly from 25 initial states before pinning control; (c) Part of state transition digraph randomly from 500 initial states after pinning control.

traditional L-based pinning control method [23]–[28], that is to change certain columns of matrix L, one needs at least controlling nodes x_1, x_2, x_5, x_6. However, using the proposed NS-based distributed pinning control method in this paper, one can achieve global stabilization by controlling nodes x_1, x_5, x_6. Without using matrix L, the computational complexity can be reduced, and the NS-based distributed pinning control method also leads to a lower dimensional controller design than the traditional L-based pinning control design.

*Example 4.2 (**T-Cell Receptor Signaling Network**):* Consider a large Boolean model of a cellular network in [35], which well model the logical dynamics of T-Cell receptor signaling model. In fact, the proposed NS-based distributed pinning control method has been tested in a network model consisting of 90 nodes denoted by $x_1, \ldots, x_{90}$ [35], the detailed logical functions can be found in Table 3 in [35] and the interaction digraph is shown in Fig. 3(a). It is said in [36], "*It is, to the best of our knowledge, the largest Boolean model of a cellular network to date*". Due to the page limitation, the corresponding logical functions are omitted here, we refer the readers to [36]. According to the proposed NS-based distributed pinning control, design an NS-based distributed pinning control just on 15 nodes x_1, x_2, x_3, x_4, x_9, x_{12}, x_{21}, x_{22}, x_{38}, x_{47}, x_{52}, $x_{68}, x_{69}, x_{78}, x_{79}$. The corresponding distributed pinning control for these 15 nodes are given below: $x_1(*) = u_1 \vee x_1, x_2(*) = u_2 \vee x_2, x_3(*) = u_3 \vee x_3, x_4(*) = u_4 \wedge x_4,$ $x_9(*) = u_9 \vee [x_2 \wedge x_4 \wedge \neg x_6 \wedge \neg x_7], x_{12}(*) = u_{12} \vee x_{17}, x_{21}(*) = u_{21} \wedge x_{21}, x_{22}(*) = u_{22} \vee x_{22}, x_{38}(*) = u_{38} \vee x_{38}, x_{47}(*) = u_{47} \vee x_{47}, x_{52}(*) = u_{52} \vee x_{49}, x_{68}(*) = u_{68} \vee x_{68}, x_{69}(*) = u_{69} \wedge x_{69},$ $x_{78}(*) = u_{78} \vee x_{78}, x_{79}(*) = u_{79} \vee x_{79}$; and the logical dynamics for each controller are given below: $u_1 = \neg x_1, u_2 = \neg x_2, u_3 = \neg x_3, u_4(*) = \neg x_4,$ $u_9 = x_2 \wedge x_4 \wedge \neg x_6,$ $u_{12} = \neg x_{17}, u_{21} = \neg x_{21},$ $u_{22} = \neg x_{22}, u_{38} = \neg x_{38}, u_{47} = \neg x_{47}, u_{52} = \neg x_{49}, u_{68} = \neg x_{68}, u_{69} = \neg x_{69}, u_{78} = \neg x_{78}, u_{79} = \neg x_{79}$. Fig. 3(b) shows part of the state transition digraph randomly from 25 initial states before pinning control, while Fig. 3(c) shows part of the state transition digraph randomly from 500 initial states after control. This implies that the proposed NS-based distributed pinning control in this paper is effective in certain biological networks, while it is almost impossible using traditional L-based pinning control [23]–[28]. This is because using the traditional L-based pinning control, one needs to calculate the

state transition matrix L with dimension $2^{90} \times 2^{90}$ in order to analyze and design pinning control strategy, which leads to an extremely high computational complexity.

Example 4.3: Another two biological networks, a reduced network of T-LGL signaling network with 18 nodes [37] and a network model of survival signaling in large granular lymphocyte leukemia with 29 nodes [38], have been tested using the proposed NS-based distributed pinning control method in this paper, which can be found in Appendix B. We refer the readers to [37], [38] for the detailed logical dynamics.

Remark 4.1: It has been formulated that gene regulatory networks in most biological systems are sparse [39], which implies that it is practical and reasonable in determining certain feasible feedback arc sets over the interaction digraph of BNs. Just as shown in [40], the mean connectivity in Escherichia coli is found between 2 and 3, which shows a rather loosely interconnected structure. Thus, the proposed edge-deleting methodology is practical and reasonable in the majority of gene networks since generally they are sparse. In addition, the proposed edge-deleting methodology can model the action of a drug that inactivates the corresponding interaction among two gene products [20], while deleting nodes of BNs can model the control action blocking of effects of products of genes associated to these nodes.

V. CONCLUSION

In this paper, a new approach for designing NS-based distributed pinning control has been presented to achieve global stabilization to any given fixed point, without using state transition matrix. The NS-based distributed pinning control is designed based on local neighbors of controlled nodes. In addition, the computational complexity is dramatically reduced, which can be efficiently applied to large-scale BNs, such as T-Cell receptor signaling Boolean model with 90 nodes. A new perspective has been proposed combining the network structure with NS-based distributed pinning control design, which aims to solve the problem of computational complexity for large-scale networks.

To sum up, by comparing the simulations on certain BNs with different sizes, the traditional L-based pinning control [23]–[28] has limitations on the following certain aspects:

I. Calculations of algebraic system $x(t+1) = Lx(t)$ is always needed, where the state transition matrix L has a

dimension $2^n \times 2^n$. Thus, when the number of network nodes n becomes larger, the computational complexity $(O(2^n \times 2^n))$ will be more complicate;

II. Since the traditional L-based pinning control is designed based on the transformation of matrix L, then L-based pinning controllers depends on all the nodes and also are in the form of $u(t) = g(x_1(t), \dots, x_n(t))$. Thus, when network nodes n become larger, L-based pinning controllers will be more complicated and have higher dimensions;

III. L-based pinning controlled nodes are determined by matrix L, when the dimension of L becomes larger, pinning controlled nodes will be difficult to be found.

The above limitations are well solved by the proposed NS-based distributed pinning control, which reduces the high computation complexity to $O(2 \times 2^K)$ (K is the largest number of neighbors among each node). By observing simulations on several biological networks, the proposed approach is effective in biological networks with sparse neighbors. In addition, the proposed pinning control technique can be applied to reduce the computational complexity of controllability and observability of BNs and BCNs.

APPENDIX

A. Example 3.1

Consider a BN with 3 nodes, and its equivalent algebraic form is given below,

$$x_1(t+1) = \hat{A}_1 x_3(t), x_2(t+1) = \hat{A}_2 x_3(t), x_3(t+1) = \delta_2^1. \quad (24)$$

Here, $\hat{A}_1 = \delta_2[2,1]$, $\hat{A}_2 = \delta_2[1,2]$. Let $x(t) = \ltimes_{j=1}^3 x_j(t)$, one has that $x(t+1) = Lx(t)$, where $L = \delta_8[5,3,5,3,5,3,5,3]$. According to Lemma 2.2, (24) is globally stable to state δ_8^5. Suppose that state δ_8^5 is not the given fixed point we want while state δ_8^7 is, here we consider how to achieve global stabilization of (24) to state δ_8^7 by transforming certain structure matrices of (24). Here, we consider Problem 3 and the constraint conditions (16) to find matrices $\check{A}_1 \triangleq \begin{pmatrix} a_{11} & a_{12} \\ 1-a_{11} & 1-a_{12} \end{pmatrix} \in \mathscr{L}_{2\times2}, \check{A}_2 \triangleq \begin{pmatrix} b_{11} & b_{12} \\ 1-b_{21} & 1-b_{22} \end{pmatrix} \in \mathscr{L}_{2\times2}, \check{A}_3 \triangleq \begin{pmatrix} c_{11} \\ 1-c_{11} \end{pmatrix} \in \mathscr{L}_{2\times1}$ and binary variables $\delta_1, \delta_2, \delta_3 \in \mathscr{D}$, such that $\mathbf{c}_3 \triangleq \sum_{i=1}^3 \delta_i$ is minimum. Thus, the constraint conditions (16) can be simplified as follows:

$$\delta_1 a_{11} = 0, \ 1-\delta_2+\delta_2 b_{11} = 0, \ 1-\delta_3+\delta_3 c_{11} = 0. \quad (25)$$

Then, Problem 3 is converted to find the indexes $a_{11}, a_{12}, b_{11}, b_{12}, c_{11} \in \mathscr{D}$, and binary variables $\delta_1, \delta_2, \delta_3 \in \mathscr{D}$, minimizing the cost function $\mathbf{c}_3 = \sum_{i=1}^3 \delta_i$ subject to (25). Thus, one can find the minimum cost $\mathbf{c}_3 = 1$ under the possible solution $\delta_1 = \delta_3 = 0$, $\delta_2 = 1$, and logical matrix $\check{A}_2$ satisfying $\mathrm{Col}_1(\check{A}_2) = \delta_2^2$. For example, there exists a feasible solution for $\check{A}_2$ as $\check{A}_2 = \delta_2[2,1]$. In fact, system (24) could be globally stabilizable to any state by transforming structure matrices according to Problem 3. For example, consider global stabilization of (24) to state δ_8^3, one can also find $\delta_1 = \delta_2 = 1$, $\delta_2 = 0$ such that $\mathbf{c}_3 = \sum_{i=1}^3 \delta_i$ equals to 2, and a feasible solution is matrices $\check{A}_1 = \delta_2[1,2]$ and $\check{A}_2 = \delta_2[2,1]$.

B. Example 4.3 (Further Applications)

A brief report is given here on applying the proposed method to reduced T-LGL signaling network with 18 nodes [37] and large granular lymphocyte leukemia with 29 nodes [38]. Consider the reduced network of T-LGL signaling network with 18 nodes, CTLA4, TCR, CREB, IFNG, P2, GPCR, SMAD, Fas, aFas, Cermide, DISC, Caspase, FLIP, BID, IAP, MCL1, S1P, Apoptosis [37]. Using the proposed NS-based distributed pinning control method in this paper, an NS-based distributed pinning controllers is designed to achieve global stabilization on nodes CTLA4, P2, DISC, S1P and Apoptosis: $\mathrm{CTLA4} = u_1 \vee \mathrm{TCR}, \mathrm{P2} = u_2 \wedge (\mathrm{IFNG} \vee \mathrm{P2}), \mathrm{DISC} = u_3 \vee [\mathrm{Cermide} \vee (\mathrm{Fas} \wedge \neg\mathrm{FLIP})], \mathrm{S1P} = u_4 \vee \neg\mathrm{Cermide}, \mathrm{Apoptosis} = u_5 \wedge (\mathrm{Caspase} \vee \mathrm{Apoptosis})$. In addition, the corresponding NS-based distributed pinning controllers are designed below: $u_1 = \neg\mathrm{TCR}, u_2 = \neg(\mathrm{IFNG} \vee \mathrm{P2}), u_3 = \mathrm{Fas} \vee (\neg\mathrm{Fas} \wedge \mathrm{Cermide}), u_4 = \mathrm{Cermide}, u_5 = \mathrm{Caspase}$.

As for the network model of survival signaling in large granular lymphocyte leukemia with 29 nodes (IL15, RAS, ERK, JAK, IL2RBT, STAT3, IFNGT, FasL, PDGF, PDGFR, PI3K, IL2, BcIxL, TPL2, SPHK, S1P, sFas, Fas, DISC, Caspase, Apoptosis, LCK, MEK, GZMB, IL2RAT, FasT, RANTES, A20, FLIP) [38], an NS-based distributed pinning controllers is designed on nodes IL15, PDGF, SPHK, in order to achieve global stabilization: $\mathrm{IL15} = u_1 \wedge \mathrm{IL15}, \mathrm{PDGF} = u_2 \vee \mathrm{PDGF}, \mathrm{SPHK} = u_3 \wedge (\mathrm{PI3K} \wedge \mathrm{S1P})$. The logical dynamics of NS-based distributed pinning controllers u_1, u_2, u_3, are $u_1 = \neg\mathrm{IL15}, u_2 = \neg\mathrm{PDGF}, u_3 = \mathrm{PI3K}$.

REFERENCES

[1] S. A. Kauffman, "Metabolic stability and epigenesis in randomly constructed genetic nets," *Journal of Theoretical Biology*, vol. 22, no. 3, pp. 437–467, 1969.

[2] S. Gupta, S. Bisht, R. Kukreti, S. Jain, and S. Brahmachari, "Boolean network analysis of a neurotransmitter signaling pathway," *Journal of Theoretical Biology*, vol. 244, no. 3, pp. 463–469, 2007.

[3] A. Saadatpour, I. Albert, and R. Albert, "Attractor analysis of asynchronous Boolean models of signal transduction networks," *Journal of Theoretical Biology*, vol. 266, no. 4, pp. 641–656, 2010.

[4] A. Melkman, X. Cheng, W. Ching, and T. Akutsu, "Identifying a probabilistic Boolean threshold network from samples," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 4, pp. 869–881, 2018.

[5] S. Azuma, T. Yoshida, and T. Sugie, "Structural monostability of activation-inhibition Boolean networks," *IEEE Transactions on Control of Network Systems*, vol. 4, no. 2, pp. 179–190, 2017.

[6] S. Azuma, T. Yoshida, and T. Sugie, "Structural oscillatory analysis of Boolean networks," *IEEE Transactions on Control of Network Systems*, vol. 6, no. 2, pp. 464–473, 2019.

[7] F. Mori and A. Mochizuki, "Expected number of fixed points in Boolean networks with arbitrary topology," *Physical Review Letters*, vol. 119, no. 2, p. 028301, 2017.

[8] D. Cheng, H. Qi, and Z. Li, *Analysis and Control of Boolean Networks: A Semi-tensor Product Approach*. Springer Science & Business Media, 2010.

[9] D. Laschov and M. Margaliot, "Controllability of Boolean control networks via Perron-Frobenius theory," *Automatica*, vol. 48, no. 6, pp. 1218–1223, 2012.

[10] S. Chen, Y. Wu, M. Macauly, and X. Sun, "Monostability and bistability of Boolean networks using semi-tensor product," *IEEE Transactions on Control of Network Systems*, vol. 6, no. 4, pp. 1379–1390, 2019.

[11] Y. Wu and T. Shen, "Policy iteration approach to control residual gas fraction in IC engines under the framework of stochastic logical dynamics," *IEEE Transactions on Control of Network Systems*, vol. 25, no. 3, pp. 1100–1107, 2017.

[12] M. Meng, J. Lam, J. Feng, and K. Cheung, "Stability and guaranteed cost analysis of time-triggered Boolean networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 8, pp. 3893–3899, 2018.

[13] I. Shmulevich, E. R. Dougherty, S. Kim, and W. Zhang, "Probabilistic Boolean networks: A rule-based uncertainty model for gene regulatory networks," *Bioinformatics*, vol. 18, no. 2, pp. 261–274, 2002.

[14] K. Zhang and L. Zhang, "Observability of Boolean control networks: A unified approach based on finite automata," *IEEE Transactions on Control of Network Systems*, vol. 61, no. 9, p. 2733–2738, 2016.

[15] D. Cheng, H. Qi, Z. Li, and J. Liu, "Stability and stabilization of Boolean networks," *International Journal of Robust and Nonlinear Control*, vol. 21, no. 2, pp. 134–156, 2011.

[16] R. Li, M. Yang, and T. Chu, "State feedback stabilization for Boolean control networks," *IEEE Transactions on Automatic Control*, vol. 58, no. 7, pp. 1853–1857, 2013.

[17] Y. Guo, P. Wang, W. Gui, and C. Yang, "Set stability and set stabilization of Boolean control networks based on invariant subsets," *Automatica*, vol. 61, pp. 106–112, 2015.

[18] R. Zhou, Y. Guo, and W. Gui, "Set reachability and observability of probabilistic Boolean networks," *Automatica*, vol. 106, pp. 230–241, 2019.

[19] J. Zanudo and R. Albert, "Cell fate reprogramming by control of intracellular network dynamics," *PLoS Computational Biology*, vol. 11, no. 4, p. e1004193, 2015.

[20] D. Murrugarra, A. Veliz-Cuba, B. Aguilar, and R. Laubenbacher, "Identification of control targets in Boolean molecular network models via computational algebra," *BMC Systems Biology*, vol. 10, no. 1, p. 94, 2016.

[21] E. Fornasini and M. E. Valcher, "On the periodic trajectories of Boolean control networks," *Automatica*, vol. 49, no. 5, pp. 1506–1509, 2013.

[22] H. Li and Y. Wang, "Further results on feedback stabilization control design of Boolean control networks," *Automatica*, vol. 83, pp. 303–308, 2017.

[23] F. Li, "Pinning control design for the stabilization of Boolean networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, pp. 1585–1590, 2015.

[24] J. Lu, J. Zhong, C. Huang, and J. Cao, "On pinning controllability of Boolean control networks," *IEEE Transactions on Automatic Control*, vol. 61, no. 6, pp. 1658–1663, 2016.

[25] F. Li, H. Yan, and H. Karimi, "Single-input pinning controller design for reachability of Boolean networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 7, pp. 3264–3269, 2018.

[26] F. Li, "Robust stabilization for a logical system," *IEEE Transactions on Control Systems Technology*, vol. 25, no. 6, pp. 2176–2184, 2017.

[27] F. Li, "Pinning control design for the synchronization of two coupled Boolean networks," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 63, no. 3, pp. 309–313, 2016.

[28] B. Li, J. Lu, Y. Liu, and Z. Wu, "The outputs robustness of Boolean control networks via pinning control," *IEEE Transactions on Control of Network Systems*, vol. 7, no. 1, pp. 201–209, 2020.

[29] Y. Zhao, B. Ghosh, and D. Cheng, "Control of large-scale Boolean networks via network aggregation," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 7, pp. 1527–1536, 2016.

[30] F. Robert, *Discrete Iterations: A Metric Study*. Springer Science & Business Media, 2012.

[31] R. Sedgewick and K. Wayne, *Algorithms*. Addison-Wesley Professional, 2011.

[32] J. Bang-Jensen and G. Gutin, *Digraphs: Theory, Algorithms and Applications*. Springer Science & Business Media, 2008.

[33] K. Kobayashi and K. Hiraishi, "Design of probabilistic Boolean networks based on network structure and steady-state probabilities," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 8, pp. 1966–1971, 2016.

[34] C. Campbell and R. Albert, "Stabilization of perturbed Boolean network attractors through compensatory interactions," *BMC Systems Biology*, vol. 8, no. 1, p. 53, 2014.

[35] Y. M. Zou, "An algorithm for detecting fixed points of Boolean network," in *2013 ICME International Conference on Complex Medical Engineering*, pp. 670–673, 2013.

[36] J. Saez-Rodriguez, L. Simeoni, J. Lindquist, R. Hemenway, U. Bommhardt, B. Arndt, U. Haus, R. Weismantel, E. Gilles, S. Klamt, *et al.*, "A logical model provides insights into t cell receptor signaling," *PLoS Computational Biology*, vol. 3, no. 8, p. e163, 2007.

[37] A. Saadatpour, R. Wang, A. Liao, X. Liu, T. Loughran, I. Albert, and R. Albert, "Dynamical and structural analysis of a T cell survival network identifies novel candidate therapeutic targets for large granular lymphocyte leukemia," *PLoS Computational Biology*, vol. 7, no. 11, p. e1002267, 2011.

[38] R. Zhang, M. V. Shah, J. Yang, S. B. Nyland, X. Liu, J. K. Yun, R. Albert, and T. P. Loughran, "Network model of survival signaling in large granular lymphocyte leukemia," *Proceedings of the National Academy of Sciences*, vol. 105, no. 42, pp. 16308–16313, 2008.

[39] H. Jeong, S. P. Mason, A.-L. Barabási, and Z. N. Oltvai, "Lethality and centrality in protein networks," *Nature*, vol. 411, no. 6833, pp. 41–42, 2013.

[40] D. Thieffry, A. M. Huerta, E. Pérez-Rueda, and J. Collado-Vides, "From specific gene regulation to genomic networks: A global analysis of transcriptional regulation in escherichia coli," *BioEssays*, vol. 20, no. 5, pp. 433–440, 1998.