Title	Block sparse design of distributed controllers for dynamical network systems
Author(s)	Adachi, Ryosuke; Yamashita, Yuh; Kobayashi, Koichi
Citation	International journal of robust and nonlinear control, 13(1), 49-66 https://doi.org/10.1002/rnc.6092
Issue Date	2023-01-10
Doc URL	http://hdl.handle.net/2115/88236
Rights	This is the peer reviewed version of the following article: Adachi, R, Yamashita, Y, Kobayashi, K. Block sparse design of distributed controllers for dynamical network systems. Int J Robust Nonlinear Control. 2023; 33(1), which has been published in final form at https://doi.org/10.1002/rnc.6092. This article may be used for non-commercial purposes in accordance with Wiley Terms and Conditions for Self-Archiving.
Type	article (author version)
File Information	main.pdf

Received: Added at production | Revised: Added at production | Accepted: Added at production

DOI: xxx/xxxx

ARTICLE TYPE

Block Sparse Design of Distributed Controllers for Dynamical Network Systems†

Ryosuke Adachi*[1] | Yuh Yamashita[2] | Koichi Kobayashi[2]

[1]Graduate School of Sciences and Technology for Innovation, Yamaguchi University, Yamaguchi, Japan
[2]Faculty of Information Science and Technology, Hokkaido University, Hokkaido, Japan

Correspondence
Ryosuke Adachi, Graduate School of Sciences and Technology for Innovation, Yamaguchi University, Ube 755-8611, Japan.Email: r-adachi@yamaguchi-u.ac.jp

Summary

This study proposes a controller design method based on block sparse optimization for dynamical network systems. The objective of the controller is to stabilize dynamical network systems with a given convergence rate. The block sparse optimization minimizes the number of controlled nodes. This study is unique in that the structure of the controller is constrained by the network topology of the system. Additionally, the proposed design problem is separable in terms of the distributed optimization over networks. The proposed method is applicable to controller design for the pinning control of consensus systems and the optimal vaccine allocation for epidemic spreading processes.

KEYWORDS:
Block Sparse, Dynamical Network System, Multi Agent System, Graph Theory

1 | INTRODUCTION

Dynamical networks are systems whose behaviors are determined by interactions over large-scale complex networks. Several phenomena, such as gene networks, electrical power networks, epidemic spreading processes, and hit phenomena, can be represented by dynamical network systems (see, e.g., References [1,2,3,4]). As the number of nodes in a network increases, substantial resources are required to design and implement controllers for dynamical network systems. Owing to the large scale of the systems, substantial computational resources are required to precisely identify a model. After identifying the model, complex and large computations are required to design the controllers. Although controllers can be designed from large and complex models, their implementation may necessitate substantial resources owing to the enormous scale of the systems. However, because these resources are limited, they cannot be utilized for designing the controllers. For instance, the spread of an epidemic can be controlled by vaccinations. However, it is not feasible for everyone to be vaccinated because the amount of vaccines is limited.

In the past decades, graph theory has been utilized to analyze dynamical systems (see, e.g., References [3,5,6,7,8]). To represent the dynamical network system based on graph theory, a structured system representation was proposed in [9]. In this representation, a directed graph depicts a dynamical system, wherein nodes correspond to states, inputs, and outputs of the original state space, and edges indicate that there exists a non-zero parameter between the endpoints of the corresponding edge in the original state space.

Because we do not require exact parameters to model structured systems, the computational cost for representing a structured system is lower than that of the original system. The properties of the graph of the structured system correlate with the generic conditions of the original system, such as controllability [10,11,8] and the number of invariant zeros [12]. However, little information is imparted on the conservative conditions of original systems. Specifically, as delineated in [9], stability conditions cannot be

†Block Sparse Design of Distributed Controllers for Dynamical Network Systems

obtained from structured systems. Because structured systems do not provide the necessary conditions, we cannot analyze stability based on network structures with a reasonable computational load.

Numerous studies have proposed the stability condition of dynamical network systems based on graph theory [13,14,15,16,17]. The stability condition of nonlinear dynamical network systems based on a feedback vertex set was proposed by [13,14]. The studies [13,14] demonstrated that the system is stable if the states corresponding to the feedback vertex set are zero. However, reference [15] indicated that the existence of a controller that stabilizes the states corresponding to the feedback vertex set cannot be guaranteed. Although other references [16,17] discuss the stability condition based on graph theory, it is confined to a positive system and a Boolean network, respectively.

In a prior study by (see Reference [18]), the authors proposed the stability condition and design methods of a controller and an observer based on graph theory and linear dynamical network systems. In this study, we utilize a directed graph with weights to represent a dynamical system based on the graph. For a directed graph with weights, to derive the stability conditions of linear dynamical systems, weighted degrees are defined. From the stability condition based on the weighted degrees, the parameter regions of the controller and observer that stabilize the system and error systems are deciphered. As an application, we consider the equilibrium point analysis of the Lotka-Volterra system. Utilizing this system, we demonstrate a condition based on the weighted degree for all species to survive. By using the framework based on weighted degrees, we can analyze the stability of a dynamical network system with a scalable computational load. However, load reduction in the design and implementation for controllers has been an open problem.

The load reduction during the implementation of controllers has been addressed in prior studies (see, e.g., Reference [19,20,21,22,23]). References [23,21] indicated that, in distributed controllers, the performance of the closed-loop system was optimized with limited resources. Because the limited resources were given as the constraints of the controller design problems, the controlled nodes were not selected by utilizing a framework [23,21]. References [19,20,22] utilized sparse optimization and geometric programming to reduce the implementation cost of the controllers, where the control performance is given as a specification. The preceding studies demonstrate potential to reduce the implementation resources for the controller. However, because exact models and centralized calculations are required, the computational cost for the design of the controller increases as the scale of the network increases.

Therefore, this study proposes an optimal resource allocation framework based on weight degrees. In the proposed framework, we utilize block sparse optimization as a tool for the design of distributed controllers. Because blocks indicate the number of controllers that need to be implemented, the proposed method can optimize the resources for the implementation of the controllers. Block sparse optimization is utilized in signal and image processing, such as compressed sensing (see, e.g., Reference [24]). The convex relaxation problem of block sparse optimization returns the same solution to the original problem if the condition based on the block isometry-restricted isometry property holds (see, e.g., References [25,26]). From this result, we expect that the convex relaxation of the proposed method provides an approximate solution. In the proposed method, the specifications of the closed-loop system are given using weighted degrees. Thus, we can solve the optimal resource allocation over the graph using the alternating direction method of multipliers (ADMM). Because the computational load of each node in the proposed method depends only on the degree of connection, the load for the design of the controller is also scalable. We present two applications of the proposed method in this paper. The first application is the design of a pinning controller with minimum controlled nodes. The second application is the optimal vaccine allocation for epidemic spreading processes. The contributions of this work are summarized as follows: 1) Our proposed method can optimize the resources for the implementation of the controller by using block sparse optimization. 2) The computational load for the design of the distributed controller is scalable.

Notation

For $A \in \mathbb{R}^{n\times n}$, $\lambda_{\max}(A)$ represents the maximum real part of the eigenvalues of A. We utilize $[A]_{i,j}$ as the (i,j)-entry of matrix A. When a is a vector, $[a]_i$ represents the ith element of a. For a finite set $\mathcal{N}$, we utilize the cardinality of $\mathcal{N}$ as $|\mathcal{N}|$. $|x|$ denotes the absolute value of x for a real number $x \in \mathbb{R}$. For a complex number $s \in \mathbb{C}$, $|s|$ denotes a norm of s. For $x := [x_1, \dots, x_n] \in \mathbb{R}^n$, the l_2 norm of x is expressed as

$$\|x\|_2 := \sqrt{\sum_{i=1}^{n} x_i^2}.$$

For the set $\mathcal{N} := \{i_1, \dots, i_n\}$, we define row $\left\{ a_i \mid i \in \mathcal{N} \right\} := [a_{i_1}, \dots, a_{i_n}]$.

2 | DISTRIBUTED CONTROLLER DESIGN BASED ON BLOCK SPARSE OPTIMIZATION

2.1 | Block Sparse Reconstruction

In this section, we introduce a block sparse reconstruction[25,26] as a preliminary segment of this work. The block sparse reconstruction returns the solution of the underdetermined systems whose l_2/l_0 norm is minimized.

Consider the following linear equation:

$$Ax = b, \tag{1}$$

where $A \in \mathbb{R}^{n\times m}$ and $b \in \mathbb{R}^m$. Vector x denotes an unknown vector and is defined as N blocks as

$$x = [x_1^\top, \dots, x_N^\top]^\top, \tag{2}$$

where $x_i \in \mathbb{R}^{n_i}$ and $\sum_{i=1}^{N} n_i = N$. Consider the solution of (1) such that the number of non-zero blocks of x is minimized. Previous works[25,26] expressed the number of non-zero blocks of x by using the l_2/l_0 norm as

$$\|x\|_{2,0} := \sum_{i=1}^{N} I(x_i)\ ,\ I(x_i) := \begin{cases} 1 & \text{if } \|x_i\|_2 \neq 0 \\ 0 & \text{otherwise.} \end{cases}$$

Then, the block sparse reconstruction of (1) is defined as

$$\begin{aligned} \text{minimize} \quad & \|x\|_{2,0} \\ \text{subject to} \quad & Ax = b. \end{aligned} \tag{3}$$

Note that we require intensive computational resources to solve the optimization problem (3) because l_2/l_0 norm $\|\cdot\|_{2,0} \to \mathbb{R}^+$ is a non-convex function for x. To reduce the computational load for solving block sparse solutions, prior studies[25,26] have also proposed a convex relaxation problem of (3). In line with prior studies[25,26], we define l_2/l_1 of x as

$$\|x\|_{2,1} := \sum_{i=1}^{N} \|x_i\|_2.$$

Note that the l_2/l_1 norm $\|\cdot\|_{2,1} \to \mathbb{R}^+$ is a convex function of x, unlike the l_2/l_0 norm. Thus, the convex optimization problem is defined as follows:

$$\begin{aligned} \text{minimize} \quad & \|x\|_{2,0} \\ \text{subject to} \quad & Ax = b. \end{aligned} \tag{4}$$

In general, convex relaxation problems do not return the same solution as that of the original problem. However, a remarkable aspect of the convex relaxation problem (4) is that (4) returns the same solution to (3) if a condition based on the block-restricted isometry property[25,26] (BRIP) holds. Let us define set $\Sigma(s)$ and BRIP as follows.

Definition 1. Assume that x denotes a vector with an N block, as in (2). For $s \in \{1, \dots, N\}$, we refer to x as s-block sparse if $\|x\|_{2,0} \leq s$. We define set $\Sigma(s)$ of a vector whose elements are s-block sparse as

$$\Sigma(s) := \{x \in \mathbb{R}^n \mid \|x\|_{2,0} \leq s\}.$$

Definition 2. Let us consider matrix $A \in \mathbb{R}^{m\times n}$. We deem that A satisfies the BRIP with parameter $\delta(s)$ if there exists $\delta(s) \in (0, 1)$ such that

$$(1 - \delta(s))\|x\|_2^2 \leq \|Ax\|_2^2 \leq (1 + \delta(s))\|x\|_2^2 \tag{5}$$

holds for any $x \in \Sigma(s)$.

Consequently, previous studies[25] have presented the following lemma.

Lemma 1. Let a block s-sparse vector hold $Ac_0 = b$. If A satisfies BRIP (5) with $\delta(2s) < \sqrt{2} - 1$, then 1) there exists a unique block s-sparse vector c that satisfies $Ac = b$; 2) The optimization problem (4) has a unique solution that is equal to the solution of (3).

2.2 | Problem Formulation

Let us consider dynamical systems over large and complex networks. A mathematical representation of the network is based on a graph-theoretic approach. A digraph is defined as a pair of $\mathcal{V}$ and $\mathcal{E}$ as $\mathcal{G} := (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is a set of nodes and $\mathcal{E}$ is a set of edges. The set of nodes is given as $\mathcal{V} := \{1, \dots, N\}$, and the set of edges is denoted by $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. The edge (i, j) indicates an interaction from the ith node to the jth node. The set of neighbors around the ith node over $\mathcal{G}$ is denoted as $\mathcal{N}_i^{\text{in}}$ and $\mathcal{N}_i^{\text{out}}$, where $\mathcal{N}_i^{\text{in}} := \{j \mid (j, i) \in \mathcal{E}\}$ is a set of in-flow neighbors, and $\mathcal{N}_i^{\text{out}} := \{j \mid (i, j) \in \mathcal{E}\}$ is a set of out-flow neighbors.

A dynamical system in which a state evolves over $\mathcal{G}$ is defined as

$$\dot{x}_i = a_i x_i + \sum_{j \in \mathcal{N}_i^{\text{in}}} w_{(j,i)} x_j + u_i \text{ for all } i \in \mathcal{V}, \tag{6}$$

where $x_i \in \mathbb{R}$ is the state and $u_i \in \mathbb{R}$ is the input of the i-th node. In (6), the self-loop a_i determines the autonomous behavior of the i-th node. In contrast, weight $w_{(i,j)}$ determines the interaction on edge (j, i). As an equivalent representation of (6), we define

$$\dot{x} = Ax + Bu, \tag{7}$$

where $x := [x_1, \dots, x_N]^\top$, $u := [u_1, \dots, u_N]^\top$, $B = I_N$ and

$$[A]_{i,j} = \begin{cases} w_{(j,i)} & (i, j) \in \mathcal{E} \\ a_i & i = j \\ 0 & \text{otherwise.} \end{cases}$$

Let us consider a distributed controller design for system (6). Suppose that the structure of the distributed controller is given by

$$u_i = k_{(i,i)} x_i + \sum_{j \in \mathcal{N}_i^{\text{in}}} k_{(j,i)} x_j. \tag{8}$$

The structure of (8) indicates that each node can utilize x_j for $j \in \{i\} \cup \mathcal{N}_i^{\text{in}}$ to calculate the control input. The feedback gain vector $k_i \in \mathbb{R}^N$ can be defined as

$$[k_i]_j = \begin{cases} k_{(j,i)} & (j, i) \in \mathcal{E} \\ k_{(i,i)} & i = j \\ 0 & \text{otherwise.} \end{cases}$$

Accordingly, (8) can be expressed by

$$u_i = k_i^\top x, \tag{9}$$

Therefore, the input for (7) is expressed as $u = Kx$, where $K = [k_1, \dots, k_N]^\top$. The closed-loop system (7) with (9) is expressed as follows:

$$\dot{x} = (A + K)x. \tag{10}$$

Thereafter, we find K by complying with the following conditions.

- Each node can utilize only states of its own and in-flow neighbors for calculating the control input.
- Real parts of eigenvalues of a closed-loop system (10) are smaller than a given upper bound.
- The number of controlled nodes, which implies $k_i \neq 0$, is minimized.

The objective of the distributed controller (9) is to stabilize the system (6). In addition, we consider the upper bound of the eigenvalues of $A + K$ as a specification of the convergence rate. Suppose that a positive value r is given for the controller design, then we find K such that $\lambda_{\max}(A + K) \leq -r$.

In compliance with the above condition, let us consider the minimization of controlled nodes. Here, we define controlled nodes as nodes that require input to achieve the control objective. In contrast, uncontrolled nodes are defined as nodes, such that $u_i = 0$ for any time period. When $u_i = 0$ for any time period, feedback gain k_i satisfies $k_i = 0$. Therefore, the number of non-zero feedback gains should be minimized to minimize the number of controlled nodes.

We formulate the above distributed controller design as the block sparse optimization problem. Let us define the feedback gain vector that includes all nodes as $k = [k_1^\top, \dots, k_N^\top]^\top$. The feedback gain vector k has N blocks, where the i-th block of k

indicates k_i. The number of non-zero blocks of k is expressed by the following l^2/l^0 norm $\|k\|_{2,0}$. Because the l^2/l^0 norm of k signifies the number of controlled nodes, we can determine the minimum design problem of the controlled nodes from the following optimization problem:

$$\begin{aligned} &\text{minimize} && \|k\|_{2,0} \\ &\text{subject to} && \lambda_{\max}(A+K) \leq -r. \end{aligned} \tag{11}$$

Because the l_2/l_0 norm is a non-convex function, we require substantial computational resources to solve (11) as the scale of $\mathcal{G}$ becomes large. Thus, we consider a convex relaxation of (11) for simplicity of calculation. In accordance with prior studies[25,26], we also utilize the l_2/l_1 norm as a convex relaxation of the l_2/l_0 norm. The convex relaxation of (11) can be expressed by

$$\text{minimize} \quad \|k\|_{2,1} \tag{12a}$$

$$\text{subject to} \quad \lambda_{\max}(A+K) \leq -r, \tag{12b}$$

In general, the convex relaxation problem (12) does not return the same solution as that of the original problem (11). However, in block sparse reconstruction, Lemma 1 indicates that the convex relaxation of the l_2/l_0 norm by the l_2/l_1 norm provides an optimal solution that is equivalent to that of the original problem with some assumptions. Thus, we expect that the convex relaxation of (11) by the l_2/l_1 norm also provides an approximate solution.

We designate r as the control specification in the controller design because we minimize the number of controlled nodes. It is assumed that there is a trade-off between the convergence rate of the closed-loop system (10) and the number of controlled nodes. For example, because $K = -A - rI$ is a candidate for the optimal solutions of (11) and (12), both optimization problems are feasible for any r. However, $K = -A - rI$ requires all nodes to be controlled. Thus, we should determine r from the worst convergence rate of the closed-loop system that needs to be realized. In contrast, we can optimize r if candidates of controlled nodes are given, as in previous work[21]. The framework proposed in this work can be extended to a controller design problem with limited resources, which will be the focus of our future work.

In this study, we solve the feedback design problem (12) by using a distributed optimization algorithm over a network. Let us define an undirected graph $\bar{\mathcal{G}} := \{\mathcal{V}, \bar{\mathcal{E}}\}$ using distributed optimization. Edge list $\bar{\mathcal{E}}$ includes (j,i) if either $(j,i) \in \mathcal{E}$ or $(i,j) \in \mathcal{E}$. The distributed optimization algorithm can provide solutions to the optimization problem over networks $\bar{\mathcal{G}}$ without integrated calculations. The optimization problem over network $\bar{\mathcal{G}}$ is defined as

$$\text{minimize} \quad f(z_1, \dots, z_N) = \sum_{i \in \mathcal{V}} f_i(z_i)\,,\ z_i := \left[z_{(i,i)}, \text{row}\left\{z_{(j,i)} \mid j \in \mathcal{N}_i^{\text{in}}\right\}\right]^\top \tag{13a}$$

$$\text{subject to} \quad \left[z_i^\top, \text{row}\left\{z_{(i,j)} \mid j \in \mathcal{N}_j^{\text{out}}\right\}\right]^\top \in \mathcal{A}_i \text{ for all } i \in \mathcal{V}, \tag{13b}$$

where f is a global objective function, z_i is a decision variable, and f_i is a local objective function of the i-th node. The global objective function f is expressed as the sum of a local objective function as (13a). Constraint (13b) indicates that the constraint of the ith node is expressed by the decision variables of the own and neighboring nodes. When the optimization problem is expressed by (13), each node can calculate the optimal solution z_i^* through the communication among neighborhoods by using the alternating direction method of multipliers[27]. Because the distributed ADMM only requires communication among neighborhoods over the network $\bar{\mathcal{G}}$, we can design feedback gains without the supervisor of the systems.

We cannot directly solve the feedback design problem (12) using a distributed optimization algorithm. The objective function (12a) can be expressed as the sum of a local objective function, namely

$$f(k_1, \dots, k_N) = \sum_{i \in \mathcal{V}} f_i(k_i)\,,\ f_i(k_i) := \|k_i\|_2. \tag{14}$$

In contrast, constraint (12b) is not formulated as (13b). We need to confirm that (12b) is satisfied from the alternative conditions that are expressed as (13b). In addition, set $\mathcal{A}_i$ for the alternative condition should be characterized by the local parameters of the system (7). Let us define the local parameters of system (7) for the ith node over $\bar{\mathcal{G}}$ as

$$\Delta_i := \left[r, a_i, \text{row}\left\{w_{(i,j)} \mid j \in \mathcal{N}_i^{\text{out}}\right\}, \text{row}\left\{w_{(j,i)} \mid j \in \mathcal{N}_i^{\text{in}}\right\}\right]^\top. \tag{15}$$

To clarify that $\mathcal{A}_i$ is characterized by Δ_i, we denote $\mathcal{A}_i(\Delta_i)$. We define the feedback gains that the ith node can obtain over $\bar{\mathcal{G}}$ as

$$K_i := \left[k_i^\top, \text{row}\left\{k_{(i,j)} \mid j \in \mathcal{N}_i^{\text{out}}\right\}\right]^\top. \tag{16}$$

Then, we consider the following problem for obtaining an alternative to (12b):

Problem 1. Network $\bar{\mathcal{G}}$ and local parameter Δ_i for $i \in \mathcal{V}$ are given. Then, we find $\mathcal{A}_i(\Delta_i)$ for all $i \in \mathcal{V}$ such that (12b) holds if $K_i \in \mathcal{A}_i(\Delta_i)$ for all $i \in \mathcal{V}$.

2.3 | Main Result

In this subsection, we present a solution of problem 1. Problem 1 explores the conditions that are required to determine the upper bound of the real part of the eigenvalue of the closed-loop system from the local information over the network $\bar{\mathcal{G}}$. We introduce the Gershgorin theorem to determine the eigenvalues of the matrix. The Gershgorin theorem states that the stability conditions of (6) can be denoted by information regarding the neighborhood over $\bar{\mathcal{G}}$, which are summarized in Lemma 2. Lemma 2 shows that $\mathcal{A}_i$ in problem 1 is given by (25), which is summarized in Theorem 2. In the final part of this subsection, we propose a distributed algorithm to calculate k_i based on the ADMM.

The Gershgorin theorem indicates the range of the complex plane in which the eigenvalues of a matrix lie. Let us define C as an $n \times n$ matrix whose entries are complex numbers. For a given matrix C, the Gershgorin disks R_i and S_i on the complex plane are defined as

$$R_i := \left\{ s \in \mathbb{C} \mid \left|s - [C]_{i,i}\right| \le \sum_{i \neq j} \left|[C]_{i,j}\right| \right\},$$

$$S_i := \left\{ s \in \mathbb{C} \mid |s - [C]_{i,i}| \le \sum_{i \neq j} |[C]_{j,i}| \right\}.$$

Then, the following theorem holds:

Theorem 1 (Gershgorin theorem)**.** Every eigenvalue of C lays the following complex plane

$$\left(\bigcup_{i=1}^{n} R_i \right) \cap \left(\bigcup_{i=1}^{n} S_i \right).$$

Based on the Gershgorin theorem, we can derive the stability conditions of (6) based on graph $\mathcal{G}$. For system (6) over graph $\mathcal{G}$, the weighted degree d_i^{in} and outdegree d_i^{out} of the ith node are defined as

$$d_i^{\text{in}} = \sum_{j \in \mathcal{N}_i^{\text{in}}} |w_{(j,i)}| \ , \ d_i^{\text{out}} = \sum_{j \in \mathcal{N}_i^{\text{out}}} |w_{(i,j)}|. \tag{17}$$

The weighted degrees are expressed as the sum of the absolute values of the weights on the edges connected to each node. Therefore, the weighted degrees represent the relation between the connections among nodes over $\mathcal{G}$. The real part of the eigenvalues of system (7) is restricted based on the weighted degree as follows:

Lemma 2. For system (7), the following equation holds:

$$\lambda_{\max}(A) \le \min\{r_{\text{in}} \ , \ r_{\text{out}}\}, \tag{18}$$

where

$$r_{\text{in}} = \max_{i \in \mathcal{V}} \left(a_i + d_i^{\text{in}}\right) \ , \ r_{\text{out}} = \max_{i \in \mathcal{V}} \left(a_i + d_i^{\text{out}}\right).$$

Proof. Gershgorin disks of matrix A in (7) are expressed as

$$R_i := \left\{ s \in \mathbb{C} \mid \left|s - a_i\right| \le d_i^{\text{in}} \right\} \ , \ S_i := \left\{ s \in \mathbb{C} \mid |s - a_i| \le d_i^{\text{out}} \right\}. \tag{19}$$

Let us denote the half-plane on the complex plane as $\Xi(d) := \{s \in \mathbb{C} \mid \text{Re}(s) \le d\}$. From (19), we find that R_i and S_i of A in (7) include $\Xi(a_i + d_i^{\text{in}})$ and $\Xi(a_i + d_i^{\text{out}})$, respectively. Then, the following inclusion relations are obtained

$$\bigcup_{i \in \mathcal{V}} R_i \subset \bigcup_{i \in \mathcal{V}} \Xi(a_i + d_i^{\text{in}}) = \Xi\left(r_{\text{in}}\right), \tag{20a}$$

$$\bigcup_{i \in \mathcal{V}} S_i \subset \bigcup_{i \in \mathcal{V}} \Xi(a_i + d_i^{\text{out}}) = \Xi\left(r_{\text{out}}\right). \tag{20b}$$

Theorems 1 and (20) indicate that the eigenvalues of system (7) lie on the following complex plane:

$$\bar{\Xi} = \Xi\left(r_{\text{in}}\right) \cap \Xi\left(r_{\text{out}}\right). \tag{21}$$

Because the maximum real parts of $\Xi\left(r_{\text{in}}\right)$ and $\Xi\left(r_{\text{out}}\right)$ are r_{in} and r_{out}, respectively, we can obtain the inclusion relation $\bar{\Xi} \subseteq \Xi(\min\{r_{\text{in}}\ ,\ r_{\text{out}}\})$. Thus, the maximum real parts on $\bar{\Xi}$ is $\min\{r_{\text{in}}\ ,\ r_{\text{out}}\}$. □

Lemma 2 shows that the upper bound of the real part of the eigenvalues is expressed by the weighted degrees. Because the weighted degrees are given by(17), d_i^{in} and d_i^{out} can be calculated from Δ_i. Thus, we can obtain $\mathcal{A}_i(\Delta_i)$ in problem 1 by applying Lemma 2 to (10). The weighted degrees for the closed-loop system (10) can be expressed as

$$d_i^{\text{in}} = \sum_{j\in\mathcal{N}_i^{\text{in}}} |w_{(j,i)} + k_{(j,i)}|\ ,\ d_i^{\text{out}} = \sum_{j\in\mathcal{N}_i^{\text{out}}} |w_{(i,j)} + k_{(i,j)}|. \tag{22}$$

Equations (22) and Lemma 2 indicate that $\lambda_{\max}(A+K)$ is bounded by the following equation:

$$\lambda_{\max}(A+K) \leq \min\{r_{\text{in}}\ ,\ r_{\text{out}}\}, \tag{23}$$

where r_{in} and r_{out} for $A+K$ are expressed by

$$r_{\text{in}} = \max_{i\in\mathcal{V}}\left(a_i + k_{(i,i)} + \sum_{j\in\mathcal{N}_i^{\text{in}}} |w_{(j,i)} + k_{(j,i)}|\right)\ ,\ r_{\text{out}} = \max_{i\in\mathcal{V}}\left(a_i + k_{(i,i)} + \sum_{j\in\mathcal{N}_i^{\text{out}}} |w_{(i,j)} + k_{(i,j)}|\right). \tag{24}$$

Thus, we can obtain $\mathcal{A}_i$ in problem 1 from the following theorem:

Theorem 2. Define $\mathcal{A}_i^{\text{in}}$ and $\mathcal{A}_i^{\text{out}}$ as

$$\mathcal{A}_i^{\text{in}} := \left\{K_i \mid a_i + k_{(i,i)} + \sum_{j\in\mathcal{N}_i^{\text{in}}} |w_{(j,i)} + k_{(j,i)}| \leq -r\right\}\ ,\ \mathcal{A}_i^{\text{out}} := \left\{K_i \mid a_i + k_{(i,i)} + \sum_{j\in\mathcal{N}_i^{\text{out}}} |w_{(i,j)} + k_{(i,j)}| \leq -r\right\}. \tag{25}$$

If $\mathcal{A}_i$ in problem 1 is given by $\mathcal{A}_i = \mathcal{A}_i^{\text{in}}$ or $\mathcal{A}_i = \mathcal{A}_i^{\text{out}}$ for all $i \in \mathcal{V}$, $A+K$ with $K_i \in \mathcal{A}_i$ satisfies (12b).

Proof. When K_i is included in $\mathcal{A}_i^{\text{in}}$ given by (25) for all $i \in \mathcal{V}$, r_{in} given by (24) satisfies $r_{\text{in}} \leq -r$. Because $\min\{r_{\text{in}}\ ,\ r_{\text{out}}\}$ is bounded by $-r$ and (23) holds, we conclude that $\lambda_{\max}(A+K) \leq -r$ if $K_i \in \mathcal{A}_i^{\text{in}}$. Similarly, we can show that $\lambda_{\max}(A+K) \leq -r$ if $K_i \in \mathcal{A}_i^{\text{out}}$. □

Theorem 2 indicates that we can confirm that either (12b) is satisfied from the local information over $\bar{\mathcal{G}}$. Based on this result, we propose distributed algorithms to calculate the controller gain using the ADMM. To apply the ADMM, we transform the controller design problem into an optimization problem expressed by the augmented Lagrangian function. Then, we present the updating law based on the ADMM as the proposed method.

Under Theorem 2, we select $\mathcal{A}_i$ as $\mathcal{A}_i^{\text{out}}$ given by (25). We can utilize ADMM if $\mathcal{A}_i$ is selected as $\mathcal{A}_i^{\text{in}}$ in the same manner. Thus, we only explain $\mathcal{A}_i = \mathcal{A}_i^{\text{out}}$. We define slack variables as $v_{(i,j)}$ and $v_{(i,i)}$, and the indicator function with respect to $\mathcal{A}_i^{\text{out}}$ as $\Phi_i(\cdot)$. The slack variables express the (i,j)-entry of $A+K$ as $v_{(j,i)} = k_{(j,i)} + w_{(j,i)}$ and $v_{(i,i)} = k_{(i,i)} + a_i$. Let us define $v_i := \left[v_{(i,i)}, \text{row}\left\{v_{(i,j)} \mid j \in \mathcal{N}_i^{\text{out}}\right\}\right]^\top$. Using v_i, $\mathcal{A}_i^{\text{out}}$ can be expressed as

$$\mathcal{A}_i^{\text{out}} := \left\{v_i \mid v_{(i,i)} + \sum_{j\in\mathcal{N}_i^{\text{out}}} |v_{(i,j)}| \leq -r\right\}.$$

Indicator function $\Phi_i(v_i)$ indicates whether $v_i \in \mathcal{A}_i^{\text{out}}$ as

$$\Phi_i(v_i) = \begin{cases} 0 & \text{if } v_i \in \mathcal{A}_i^{\text{out}} \\ \infty & \text{otherwise.} \end{cases}$$

Then, we can transform the controller design problem (12) into the following optimization problem:

$$\begin{aligned} &\text{minimize} && \sum_{i\in\mathcal{V}} \left(\|k_i\|_2 + \Phi_i(v_i)\right) \\ &\text{subject to} && v_{(j,i)} = k_{(j,i)} + w_{(j,i)} \text{ for all } (j,i) \in \mathcal{E}\ ,\ v_{(i,i)} = k_{(i,i)} + a_i \text{ for all } i \in \mathcal{V}. \end{aligned} \tag{26}$$

We define the augmented Lagrangian function for (26) as $L := \sum_{i\in\mathcal{V}} L_i(k_i, v_i, y_i)$, where the local augmented Lagrangian function $L_i(k_i, v_i, y_i)$ is expressed as

$$L_i(k_i, v_i, y_i) := \|k_i\|_2 + \Phi_i(v_i) + \sum_{j\in\mathcal{N}_i^{\mathrm{out}}} \left(y_{(i,j)}(k_{(i,j)} + w_{(i,j)} - v_{(i,j)}) + \frac{\rho}{2}\|k_{(i,j)} + w_{(i,j)} - v_{(i,j)}\|_2^2 \right) + y_{(i,i)}(k_{(i,i)} + a_i - v_{(i,i)}) + \frac{\rho}{2}\|k_{(i,i)} + a_i - v_{(i,i)}\|_2^2,$$

$$y_i := \left[y_{(i,i)}, \mathrm{row}\left\{ y_{(i,j)} \mid j \in \mathcal{N}_i^{\mathrm{out}} \right\} \right]^\top.$$

In the local augmented Lagrangian function, y_i indicates a dual variable, and ρ indicates a penalty parameter.

ADMM is an iterative algorithm for solving the minimization problem of the augmented Lagrangian function. When the original optimization problem is formulated as (26), each variable is updated from the information in the neighborhood over $\bar{\mathcal{G}}$. Let us define a clock of the network as τ. Note that τ represents a discrete-time system, although system (7) is expressed as a continuous-time system. Because we consider some physical systems as motivating examples of system (7), we represent the dynamics by using the continuous-time system. In contrast, the ADMM algorithm is executed in cyberspace. Therefore, we utilize discrete-time representation to denote the ADMM in this study. The variables that the ith node can update are k_i, v_i, and y_i. To update these variables, the ith node receives message $v^\tau_{(j,i)}$, $y^\tau_{(j,i)}$, and $k^\tau_{(i,j)} + w_{(i,j)}$ from the jth node. When the main variables are updated in the ADMM, the other variables are fixed. Then, we determine the main variables that minimize the augmented Lagrangian function as follows:

$$\begin{aligned} &\text{find} \quad k_i \\ &\text{minimize} \quad \|k_i\|_2 + \sum_{j\in\mathcal{N}_i^{\mathrm{in}}} \left(y^\tau_{(j,i)}(k_{(j,i)} + w_{(j,i)} - v^\tau_{(j,i)}) + \frac{\rho}{2}\|k_{(j,i)} + w_{(j,i)} - v^\tau_{(j,i)}\|_2^2 \right) \\ &\qquad\qquad + y^\tau_{(i,i)}(k_{(i,i)} + a_i - v^\tau_{(i,i)}) + \frac{\rho}{2}\|k_{(i,i)} + a_i - v^\tau_{(i,i)}\|_2^2 \end{aligned} \tag{27}$$

Next, we fix the main and dual variables to update the slack variables. Then, we identify the slack variables that minimize the augmented Lagrangian function as follows:

$$\begin{aligned} &\text{find} \quad v_i \\ &\text{minimize} \quad \sum_{j\in\mathcal{N}_i^{\mathrm{out}}} \left(y^\tau_{(i,j)}(k^{\tau+1}_{(i,j)} + w_{(i,j)} - v_{(i,j)}) + \frac{\rho}{2}\|k^{\tau+1}_{(i,j)} + w_{(i,j)} - v_{(i,j)}\|_2^2 \right) \\ &\qquad\qquad + y^\tau_{(i,i)}(k^{\tau+1}_{(i,i)} + a_i - v_{(i,i)}) + \frac{\rho}{2}\|k^{\tau+1}_{(i,i)} + a_i - v_{(i,i)}\|_2^2 \\ &\text{subject to} \quad v_i \in \mathcal{A}_i^{\mathrm{out}} \end{aligned} \tag{28}$$

The update of dual variables is based on a gradient method as follows:

$$y^{\tau+1}_{(i,j)} = y^\tau_{(i,j)} + \rho(k^{\tau+1}_{(i,j)} + w_{(i,j)} - v^{\tau+1}_{(i,j)}), \; y^{\tau+1}_{(i,i)} = y^\tau_{(i,i)} + \rho(k^{\tau+1}_{(i,i)} + a_i - v^{\tau+1}_{(i,i)}). \tag{29}$$

In summary, each node executes the following steps to calculate the controller gain k_i.

Step 1 Each node receives $y^\tau_{(j,i)}$ and $v^\tau_{(j,i)}$ from neighborhood nodes, and calculates $k_i^{\tau+1}$ from (27).

Step 2 Each node receives $k^{\tau+1}_{(i,j)} + w_{(i,j)}$ from neighborhood nodes, and calculates $v_i^{\tau+1}$ from (28).

Step 3 Each node calculates $y_i^{\tau+1}$ from (29).

Theorem 2 stipulates a sufficient condition that states that $A + K$ is a stable matrix, although we can design feedback gains in a distributed manner. Theorem 2 stipulates the sufficient condition because the Gerhgorin theorem is a conservative result for inferring the spectrum of the matrices. We require a more accurate algorithm to estimate the spectrum and mitigate the conservativeness of this work. Some studies have proposed an estimation method for the spectrum of the matrices by using local information over the graph. For instance, the algorithm proposed by Chen et al.[28] infers the spectrum of adjacency matrices by using the degrees and count of subgraphs. Because these algorithms can only infer the adjacency matrices, we cannot utilize them for controller design, as delineated in this study. Extensions of these studies to the design of controllers form the scope of our future work.

3 | APPLICATION

3.1 | Pinning Controller Design Based on Block Sparse Optimization

Let us consider a consensus system with external inputs. The consensus system over directed graph $\mathcal{G}$ is expressed as

$$\dot{x}_i = -\sum_{i\in\mathcal{N}_j^{\text{in}}} (x_i - x_j), \tag{30}$$

where $x_i \in \mathbb{R}$ is a state of the ith node. Assume that $\mathcal{G}$ is a connected graph. Then, system (30) achieves an average consensus as $\lim_{t\to\infty} x_i = x_{\text{con}} := \sum_{i\in\mathcal{V}} x_i(0)/|\mathcal{V}|$ for all $i \in \mathcal{V}$. We define pinning nodes as nodes that are applied with an external input as

$$\dot{x}_i = -\sum_{i\in\mathcal{N}_j^{\text{in}}} (x_i - x_j) + u_i \ \text{ if } i \in \mathcal{P}, \tag{31}$$

where $\mathcal{P} \subseteq \mathcal{V}$ represents a set of pinning nodes. The objective of input u_i for all $i \in \mathcal{P}$ is to change the consensus values from x_{con} to a given value x_{pin}.

Let us consider the problem of designing a controller for u_i and the selection of $\mathcal{P}$ in (31). To calculate the input u_i, the pinning node can utilize $\bar{x}_i := \left[x_i, \text{row}\left\{x_j \mid j \in \mathcal{N}_i^{\text{in}}\right\}\right]^\top$ and a new consensus value x_{pin}. Thus, a structure of the controller for u_i is given by

$$u_i = c_i(\bar{x}_i, x_{\text{pin}}). \tag{32}$$

The controller $c_i(\cdot)$ for all $i \in \mathcal{P}$ is designed such that $\lim_{t\to\infty} x_i = x_{\text{pin}}$ with speed $-r < 0$, implying that

$$\|\eta\|_2 \le \beta\|\eta(0)\|_2 e^{-rt}\ ,\ \eta^\top := [\eta_i, \dots, \eta_N]\ ,\ \eta_i = x_i - x_{\text{pin}}.$$

The pinning nodes are selected such that the number of nodes $|\mathcal{P}|$ is minimized. Then, the above design problem is summarized as

Problem 2. For systems (30) and (31), the structure of the controller (32), consensus value x_{pin}, and convergence rate $-r$ are given. Then, we determine $c_i(\cdot)$ and $\mathcal{P}$ such that

$$\begin{aligned} &\text{minimize} \quad |\mathcal{P}| && (33a)\\ &\text{subject to} \quad \|\eta\|_2 \le \beta\|\eta(0)\|_2 e^{-rt}. && (33b)\end{aligned}$$

Let us consider the stabilization of $\dot{\eta}$ based on Theorem 2 to satisfy (33b). Because the dynamics of x_i are expressed by (30) or (31), $\dot{\eta}_i$ can be expressed as

$$\dot{\eta}_i = -|\mathcal{N}_i^{\text{in}}|\eta_i + \sum_{j\in\mathcal{N}_i^{\text{in}}} \eta_j + u_i. \tag{34}$$

We cannot directly express the dynamics of η_i as (34) for $i \notin \mathcal{P}$ because $\dot{x}_i$ is expressed by (30). However, we can consider that (30) represents the dynamics of the pinning nodes with external input $u_i = 0$ for any time period. Therefore, we also express $\dot{\eta}_i$ for all $i \notin \mathcal{P}$ as a system with $u_i = 0$ for any time period. Following the problem formulation in Section 2.2, we consider the controller whose structure is expressed by (8) as follows:

$$u_i = k_{(i,i)}\eta_i + \sum_{j\in\mathcal{N}_i^{\text{in}}} k_{(j,i)}\eta_j. \tag{35}$$

Because $k_i = 0$ for all $i \notin \mathcal{P}$, the number of pinning nodes can be expressed by $\|k\|_{2,0}$. Thus, the convex relation of (33a) can be expressed as $\sum_{i\in\mathcal{V}} \|k_i\|_2$. Set $\mathcal{A}_i^{\text{out}}$ as

$$\mathcal{A}_i^{\text{out}} := \left\{ K_i \ \middle|\ -|\mathcal{N}_i^{\text{in}}| + k_{(i,i)} + \sum_{j\in\mathcal{N}_i^{\text{out}}} |1 + k_{(i,j)}| \le -r \right\}. \tag{36}$$

Theorem 2 indicates that if K_i is included in $\mathcal{A}_i^{\text{out}}$ defined by (36), (33b) holds. Therefore, the following optimization problem is a convex relaxation of (33):

$$\begin{aligned} &\text{minimize} \quad \sum_{i\in\mathcal{V}} \|k_i\|_2 \\ &\text{subject to} \quad K_i \in \mathcal{A}_i^{\text{out}} \quad \text{defined by (36).} \end{aligned} \tag{37}$$

Then, we can obtain the following theorem:

Theorem 3. Let k_i be the optimal solution of (37). Then, the following $\mathcal{P}$ and $c_i(\cdot)$ satisfy the convex relaxation of (33):

$$\mathcal{P} = \{i \mid k_i \neq 0\}, \tag{38a}$$

$$c_i(\bar{x}_i, x_{\text{pin}}) = k_{(i,i)}(x_i - x_{\text{pin}}) + \sum_{j\in\mathcal{N}_i^{\text{in}}} k_{(j,i)}(x_j - x_{\text{pin}}). \tag{38b}$$

Proof. Based on the above discussion, (37) is the convex relaxation of (33). Because the pinning nodes are given by the node whose feedback gain in the controller is not zero, $\mathcal{P}$ is obtained as (38a). By substituting $\eta_i = x_i - x_{\text{pin}}$ into (35), we obtain

$$u_i = k_{(i,i)}(x_i - x_{\text{pin}}) + \sum_{j\in\mathcal{N}_i^{\text{in}}} k_{(j,i)}(x_j - x_{\text{pin}}).$$

Because u_i is expressed by (32), (38b) is obtained. □

3.2 | Optimal Vaccine Allocation Problem for Epidemic Spreading Processes

Let us consider networked susceptible-infected-removed (SIR) models to represent epidemic spreading processes over $\mathcal{G}$, where we assume that $\mathcal{G}$ is an undirected graph. In the networked SIR model, each node can be in one of three states: susceptible, infected, or removed. Because susceptible nodes have no immunities, they may become infected if they are in contact with infected nodes. The infected nodes become removed after they gain the immunities. Thus, the epidemic spreading process based on an SIR model is expressed as

$$\dot{S}_i = -\beta_i S_i \sum_{j\in\mathcal{N}_i^{\text{in}}} I_j \tag{39a}$$

$$\dot{I}_i = \beta_i S_i \sum_{j\in\mathcal{N}_i^{\text{in}}} I_j - \gamma_i I_i \tag{39b}$$

$$\dot{R}_i = \gamma_i I_i, \tag{39c}$$

where $\beta_i > 0$ and $\gamma_i > 0$ denote infection and recovery rates. In (39), S_i, I_i, and $R_i \in [0, 1]$ express probabilities such that node i is susceptible, infected, or removed, respectively. Thus, $S_i + I_i + R_i = 1$ holds.

Let us consider an optimal vaccine allocation problem for stabilizing (39). Note that (39) is a positive system and $R_i \geq R_i(0)$ holds for all $i \in \mathcal{N}$ because S_i, I_i, and $R_i \geq 0$ hold for all $i \in \mathcal{N}$. Thus, by (39b) we obtain

$$\dot{I}_i \leq -\gamma_i I_i + \beta_i \sum_{j\in\mathcal{N}_i^{\text{in}}} I_j + u_i \tag{40a}$$

$$u_i = -\beta_i R_i(0) \sum_{j\in\mathcal{N}_i^{\text{in}}} I_j \tag{40b}$$

for all $i \in \mathcal{N}$. In general, it is assumed that $R_i(0) = 0$ for all $i \in \mathcal{N}$ because no one gains immunity. However, the nodes can gain immunity after vaccination, which indicates that we design $R_i(0)$ by using the vaccines. Let us design a feedback controller (40b), which stabilizes (39b) by selecting $R_i(0)$. Define k_i and I as

$$I = [I_1, \dots, I_N]^\top, \ [k_i]_j = \begin{cases} -\beta_i R_i(0) & (i,j) \in \mathcal{E} \\ 0 & \text{otherwise} \end{cases}, \ [A]_{i,j} = \begin{cases} -\gamma_i & i = j \\ \beta_i & (i,j) \in \mathcal{E} \\ 0 & \text{otherwise.} \end{cases} \tag{41}$$

Then, (40b) and (40) can be expressed as $u_i = k_i^\top I$ and

$$\dot{I}(t) \leq AI(t) + Bu(t)\ ,\ u(t) = KI(t).$$

Hence, we can obtain the following equation from the comparison principle[29]:

$$I(t) \leq \exp\left((A+K)t\right) I(0). \tag{42}$$

Thus, we can stabilize (39b) by selecting $R_i(0)$ such that $A+K$ becomes a stable matrix. We then minimize the vaccinations for the stabilization of (39b). We can evaluate the amount of vaccination as

$$\sum_{i=1}^{N} I(R_i(0)). \tag{43}$$

In summary, the optimal vaccine allocation problem is formulated as follows.
Problem 3. For system (39b), the structure of controller (40b) is given. Then, we determine $R_i(0)$ for all $i \in \mathcal{V}$ such that

$$\begin{aligned} &\text{minimize} && \sum_{i=1}^{N} I(R_i(0)) && \text{(44a)} \\ &\text{subject to} && \lambda_{\max}(A+K) \leq 0 && \\ & && 0 \leq R_i(0) \leq 1 \text{ for all } i \in \mathcal{V}. && \text{(44b)} \end{aligned}$$

Let us solve problem 3 by using Theorem 2. According to (16) and (25), K_i and $\mathcal{A}_i^{\text{out}}$ for (40) are given by

$$K_i := \left[k_i^\top, \text{row}\left\{\beta_j R_i(0) \mid j \in \mathcal{N}_i^{\text{out}}\right\}\right]^\top . \mathcal{A}_i^{\text{out}} = \left\{ K_i \middle| -\gamma_i + \sum_{j\in\mathcal{N}_i^{\text{in}}} \beta_j (1 - R_j(0)) \leq 0 \right\}. \tag{45}$$

Let us consider the following optimization problem

$$\begin{aligned} &\text{minimize} && \sum_{i\in\mathcal{V}} \|k_i\|_2 && \text{(46a)} \\ &\text{subject to} && K_i \in \mathcal{A}_i^{\text{out}} \text{ defined by (45)} && \\ & && 0 \leq R_i(0) \leq 1 \text{ for all } i \in \mathcal{V}. && \text{(46b)} \end{aligned}$$

Then, we can obtain the following theorem:

Theorem 4. The optimization problem (46) is a convex relaxation of (44), and we can obtain $R_i(0)$ that satisfies (44b) by solving (46).

Proof. Because k_i is defined by (41), $\|k_i\|_2 = \beta_i R_i(0)|\mathcal{N}_i^{\text{in}}|$ holds, indicating that $\|k_i\|_2 = 0$ if and only if $R_i(0) = 0$. Thus, the amount of vaccination can be expressed as

$$\sum_{i=1}^{N} I(R_i(0)) = \|k\|_{2,0}.$$

Hence, (46a) is a convex relaxation of (44a). Theorem 2 indicates that (44b) holds if (46b). □

4 | NUMERICAL SIMULATION

4.1 | Pinning Controller Design

In this section, a numerical simulation is performed to demonstrate that Theorem 3 solves problem 2. The pinning node design shown in problem 2 is an application of problem 1. We consider pinning control over the network illustrated in Fig. 1. By solving (37) using distributed ADMM (27)–(29), we can obtain the feedback gains illustrated in Fig. 2). We can confirm that the pinning node changes the consensus value from Fig. 5).

Let us consider the consensus system and pinning controller expressed by (30) and (32), respectively. The network utilized in (30) and (32) is a scale-free network, as shown in Fig. 1). We generate the network illustrated in Fig. 1 using the Barabási-Albert (BA) model[30]. The new consensus value x_{pin} and convergence rate r are given by $x_{\text{pin}} = 1$ and $r = 0.92$, respectively.

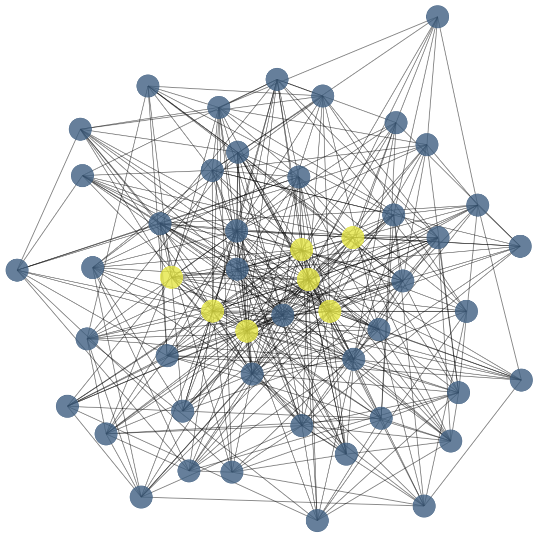

FIGURE 1 Scale-free networks generated by BA model. Yellow cycles denote the pinning node determined by (38a).

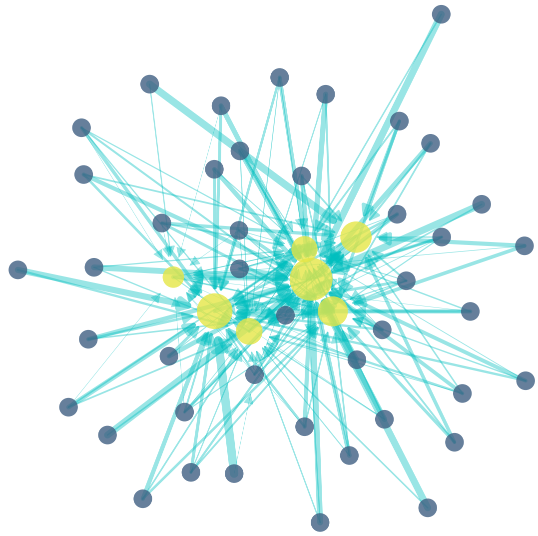

FIGURE 2 Network C represents feedback gains in (38b). The width of each edge is proportional to $|k_{(j,i)}|$. The yellow cycles represent the pinning nodes, and the size of each node is proportional to $|k_i|$.

Based on the above setting, we design a pinning controller using Theorem 3. The optimization problem (37) is solved by using the distributed ADMM expressed by (27)–(29) with $\rho = 0.5$. Figure 3 shows the l_2/l_1 norm solved by the distributed ADMM and the centralized calculation using IBM CPLEX. We can confirm that our proposed algorithm can calculate the optimal l_2/l_1 norm of the feedback gains. Figure 4 shows the l_2 norm of the feedback gains in each node, where the yellow and blue lines denote the pinning and other nodes, respectively. The optimal solutions plotted in Fig. 4 represent the l_2 norm of the pinning nodes calculated from the centralized calculation using IBM CPLEX. We can also confirm that the proposed algorithm can calculate the feedback gains for each node. Based on (38a), we determine the set of pinning nodes from Fig. 4). Fig. 4 indicates that seven nodes are selected as the pinning nodes because $k_i \neq 0$. The nodes selected as the pinning nodes are plotted as yellow cycles in Fig. 1). The feedback gains obtained from (37) are illustrated in Fig. 2). Let us define $C := (\mathcal{V}, \mathcal{E}_c)$ as the directed graph illustrated in Fig. 2, where $\mathcal{E}_c := \{(i, j) \mid k_{(i,j)} \neq 0, (i, j) \in \mathcal{E}\}$. The width of each edge in Fig. 2 is proportional to $|k_{(j,i)}|$. The yellow cycles shown in Fig. 2 represent the pinning nodes, and the size of each node is proportional to $|k_i|$.

Set $k_{(j,i)}$ is calculated from (37) to the gains in (38b). We designate $x_i(0)$ for all $i \in \mathcal{V}$ as shown in Fig. 5. Fig. 5 depicts the simulation result of system 3.1 with controller (38b), where the yellow and blue lines represent the time response of the pinning and other nodes, respectively. If $u_i = 0$, all the states in Fig. 5 converge to $x_{\rm con}$. However, we confirm that the pinning nodes change from $x_{\rm con}$ to $x_{\rm in}$. The maximum eigenvalue of the closed-loop system in this simulation becomes $\lambda_{\max} = -0.9082$. $\lambda_{\max} > r = -0.92$ because we calculate the feedback gains by using the distributed ADMM. If we directly solve (37) by using a centralized solver such as IBM CPLEX, we can obtain the exact solution that satisfies (12b).

4.2 | Optimal Vaccine Allocation Problem

In this subsection, we present a numerical simulation of the optimal vaccine allocation using (46). Consider a networked SIR model over a scale-free network with 200 nodes, as illustrated in Fig. 6. We generated the network illustrated in Fig. 6 using the BA models. We randomly generate the infection rate β_i and recovery rate γ_i. In the numerical simulation without the vaccinations, we set $R_i(0) = 0$, $I_i(0) = 0$, and $S_i(0) = 1$ for all $i \in \mathcal{V} \setminus \{1\}$. We then specify the initial states of node 1 as $R_i(0) = 0$, $I_i(0) = 0.01$, and $S_i(0) = 0.99$. Thus, Fig. 7 indicates a time response of $I(t)$. From Fig. 7, we can confirm an increase in infected nodes if there exist no nodes with immunity. To prevent the increase in infected nodes, we consider the optimal vaccine allocation using (46). Fig. 8 indicates a time response of $I(t)$, where some nodes take the vaccinations. In contrast to Fig. 7, we can confirm that the vaccinations prevent an increase in infected nodes from Fig. 8. The initial states of the removed nodes

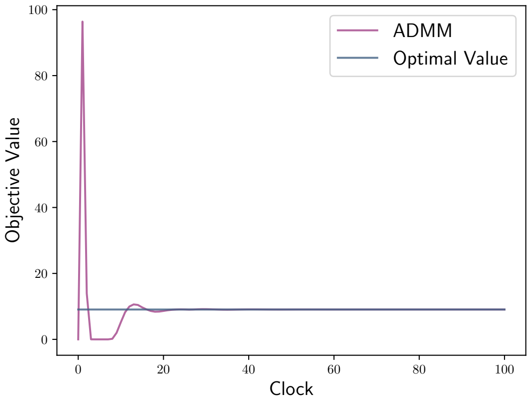

FIGURE 3 Time response of objective function value. The optimal value in this figure indicates the value of the objective function solved by the centralized calculation.

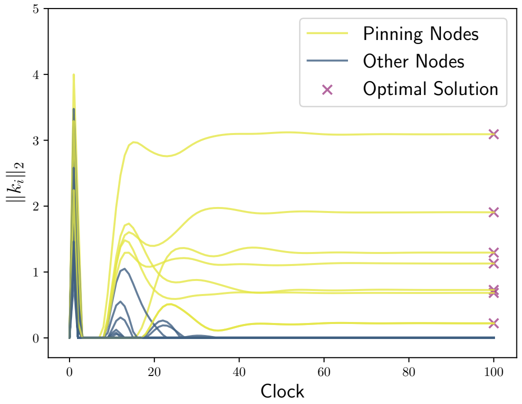

FIGURE 4 Time response of $\|k_i\|_2$ solved using distributed ADMM

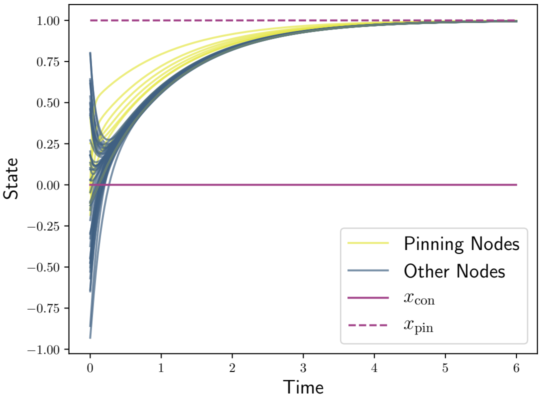

FIGURE 5 Time response of consensus system with pinning controller

$R_i(0)$ calculated from the optimization problem (46) are shown in Fig. 6, where the size of each node is proportional to $R_i(0)$. In this simulation, we obtain $\sum_{i\in\mathcal{V}} R_i(0) = 94.521$ from the vaccine allocation problem (46), which indicates that 47% of the nodes should receive vaccinations.

5 | CONCLUSION

This study applies block sparse optimization to design the controllers of dynamical network systems. Block sparse optimization can minimize the number of controlled nodes with the given convergence rates. A remarkable feature of this result is that 1) the structure of the controller is constrained by the network topology of the system, and 2) the proposed design problem is convex and separable in terms of a distributed optimization over networks. As an application, this study presents a controller design for the pinning control of consensus systems.

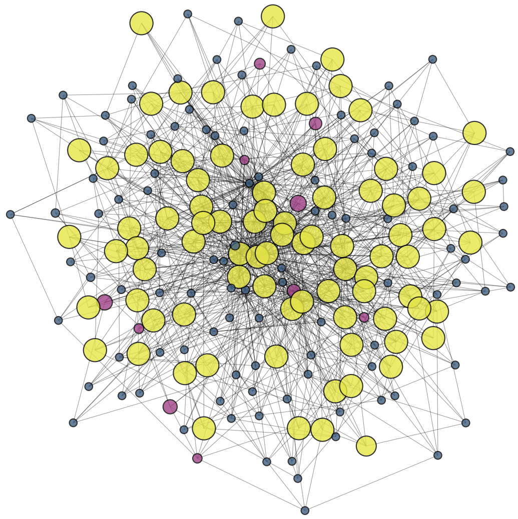

FIGURE 6 Scale-free networks generated using the BA model. The size of each node is proportional to $R_i(0)$. The yellow cycles indicate that $0.9 < R_i(0) \leq 1$. The red cycles indicate that $0.5 < R_i(0) \leq 0.9$. The blue cycles indicate that $0.0 < R_i(0) \leq 0.5$.

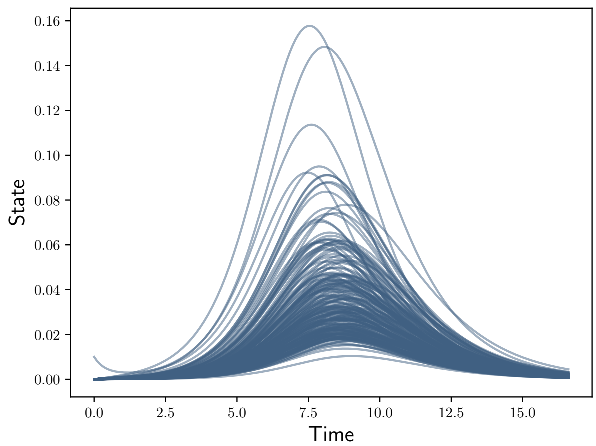

FIGURE 7 Time response of $I(t)$, where $R_i(0) = 0$ for all $i \in \mathcal{V}$.

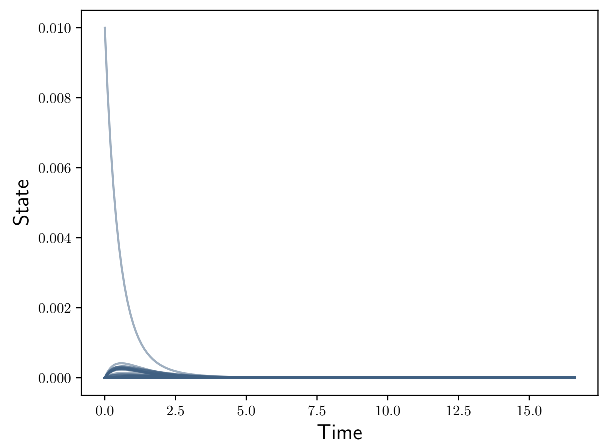

FIGURE 8 The time response of $I(t)$, where $R_i(0)$ for all $i \in \mathcal{V}$ is given as the solution to (46).

ACKNOWLEDGMENTS

This work was partially supported by JSPS KAKENHI Grant Numbers JP17K06486, JP19H02157, and JP20K14765.

References

1. Boccaletti S, Latora V, Moreno Y, Chavez M, Hwang DU. Complex networks: Structure and dynamics. *Physics reports* 2006; 424(4-5): 175–308.
2. Mesbahi M, Egerstedt M. *Graph theoretic methods in multiagent networks*. 33. Princeton University Press . 2010.

3. Liu YY, Slotine JJ, Barabási AL. Controllability of complex networks. *Nature* 2011; 473(7346): 167.
4. Ishii A, Arakaki H, Matsuda N, et al. The 'hit' phenomenon: a mathematical model of human dynamics interactions as a stochastic process. *New journal of physics* 2012; 14(6): 063018.
5. Wang LZ, Su RQ, Huang ZG, et al. A geometrical approach to control and controllability of nonlinear dynamical networks. *Nature communications* 2016; 7: 11323.
6. Gu S, Pasqualetti F, Cieslak M, et al. Controllability of structural brain networks. *Nature communications* 2015; 6: 8414.
7. Liu YY, Slotine JJ, Barabási AL. Observability of complex systems. *Proceedings of the National Academy of Sciences* 2013; 110(7): 2460–2465.
8. Menara T, Bassett DS, Pasqualetti F. Structural controllability of symmetric networks. *IEEE Transactions on Automatic Control* 2018; 64(9): 3740–3747.
9. Dion JM, Commault C, Van Der Woude J. Generic properties and control of linear structured systems: a survey. *Automatica* 2003; 39(7): 1125–1144.
10. Glover K, Silverman L. Characterization of structural controllability. *IEEE Transactions on Automatic control* 1976; 21(4): 534–537.
11. Shields R, Pearson J. Structural controllability of multiinput linear systems. *IEEE Transactions on Automatic control* 1976; 21(2): 203–212.
12. Woude v. dJ. The generic number of invariant zeros of a structured linear system. *SIAM Journal on Control and Optimization* 1999; 38(1): 1–21.
13. Mochizuki A, Fiedler B, Kurosawa G, Saito D. Dynamics and control at feedback vertex sets. II: A faithful monitor to determine the diversity of molecular activities in regulatory networks. *Journal of theoretical biology* 2013; 335: 130–146.
14. Fiedler B, Mochizuki A, Kurosawa G, Saito D. Dynamics and control at feedback vertex sets. I: Informative and determining nodes in regulatory networks. *Journal of Dynamics and Differential Equations* 2013; 25(3): 563–604.
15. Zañudo JGT, Yang G, Albert R. Structure-based control of complex networks with nonlinear dynamics. *Proceedings of the National Academy of Sciences* 2017; 114(28): 7234–7239.
16. Ogura M, Preciado VM. Optimal design of switched networks of positive linear systems via geometric programming. *IEEE Transactions on Control of Network Systems* 2017; 4(2): 213–222.
17. Azuma S, Yoshida T, Sugie T. Structural monostability of activation-inhibition Boolean networks. *IEEE Transactions on Control of Network Systems* 2017; 4(2): 179–190.
18. Adachi R, Yamashita Y, Kobayashi K. Design of Controller and Observer for Dynamical Network Systems Based on Weighted Degrees. *Proc. of the 21st IFAC World Congress* 2020: 3392–3397.
19. Polyak B, Khlebnikov M, Shcherbakov P. An LMI approach to structured sparse feedback design in linear control systems. *Proc. of 2013 European control conference (ECC)* 2013: 833–838.
20. Nagahara M, Ogura M, Yamamoto Y. Iterative greedy LMI for sparse control. *IEEE Control Systems Letters* 2021; 6: 986–991.
21. Torres JA, Roy S, Wan Y. Sparse resource allocation for linear network spread dynamics. *IEEE Transactions on Automatic Control* 2016; 62(4): 1714–1728.
22. Ogura M, Preciado VM. Efficient containment of exact SIR Markovian processes on networks. In: IEEE. ; 2016: 967–972.
23. Tsumura K, Kawasaki I. Optimal control/observation points problem and separation principle of weakly controlled large-scaled multi-agent systems. *Proc. of IEEE 55th Conference on Decision and Control (CDC)* 2016: 5110–5115.

24. Wang Y, Tang YY, Li L, Zheng X. Block sparse representation for pattern classification: Theory, extensions and applications. *Pattern Recognition* 2019; 88: 198–209.

25. Eldar YC, Mishali M. Robust recovery of signals from a structured union of subspaces. *IEEE Transactions on Information Theory* 2009; 55(11): 5302–5316.

26. Ranjan S, Vidyasagar M. Tight performance bounds for compressed sensing with conventional and group sparsity. *IEEE Transactions on Signal Processing* 2019; 67(11): 2854–2867.

27. Boyd S, Parikh N, Chu E. *Distributed optimization and statistical learning via the alternating direction method of multipliers*. Now Publishers Inc . 2011.

28. Chen X, Ogura M, Preciado VM. Bounds on the spectral radius of digraphs from subgraph counts. *SIAM Journal on Matrix Analysis and Applications* 2020; 41(2): 525–553.

29. Kirkilionis M, Walcher S. On comparison systems for ordinary differential equations. *Journal of mathematical analysis and applications* 2004; 299(1): 157–173.

30. Albert R, Barabási AL. Statistical mechanics of complex networks. *Reviews of modern physics* 2002; 74(1): 47.

How to cite this article: R. Adachi, Y. Yamashita, and K. Kobayashi (2021), A regime analysis of Atlantic winter jet variability applied to evaluate HadGEM3-GC2, *Q.J.R. Meteorol. Soc.*, *2017;00:1–6*.