

Received January 26, 2020, accepted February 5, 2020, date of publication February 17, 2020, date of current version February 27, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2974243

Analysis and Design of Multi-Agent Systems in Spatial Frequency Domain: Application to Distributed Spatial Filtering in Sensor Networks

SHINSAKU IZUMI¹, (Member, IEEE), SHUN-ICHI AZUMA², (Senior Member, IEEE), AND TOSHIHARU SUGIE³, (Fellow, IEEE)

¹Faculty of Computer Science and Systems Engineering, Okayama Prefectural University, Okayama 719-1197, Japan

²Graduate School of Engineering, Nagoya University, Nagoya 464-8603, Japan

³Graduate School of Engineering, Osaka University, Osaka 565-0871, Japan

Corresponding author: Shinsaku Izumi (izumi@cse.oka-pu.ac.jp)

This work was supported in part by the Japan Science and Technology Agency, Core Research for Evolutional Science and Technology, Japan, under Grant JPMJCR15K1, and in part by the Japan Society for the Promotion of Science, Grants-in-Aid for Scientific Research, Japan, under Grants 25249058, 17H03280, and 19K15016.

ABSTRACT This study attempts to analyze and design multi-agent systems in the spatial frequency domain and demonstrates that the spatial frequency-based approach is useful for distributed spatial filtering in sensor networks. First, we take the consensus of multi-agent systems (*i.e.*, letting the states of all agents converge to an identical value) as an example and analyze it using the concept of spatial frequencies. We then show that consensus by typical controllers corresponds to lowpass filtering in the spatial frequency domain. This demonstrates that spatial frequencies can characterize the behavior of multi-agent systems. Second, we present a controller design method in the spatial frequency domain. The designed controllers provide the feedback system with a desired spatial frequency characteristic given in advance. We further derive a sufficient condition for the spatial frequency characteristic to ensure that the designed controllers are distributed. Finally, the effectiveness and applicability of our design method are demonstrated through an example of distributed denoising in a sensor network.

INDEX TERMS Distributed control, graph signal processing, multi-agent systems, sensor networks, spatial frequency domain.

I. INTRODUCTION

The control of multi-agent systems has been an active research topic in the systems and control field. This is because multi-agent control is necessary for realizing a variety of modern systems, such as unmanned aerial vehicles, sensor networks, and smart grids [1]. In fact, these systems consist of numerous agents (corresponding to subsystems) and require control techniques for achieving coordination among them.

To date, multi-agent control has been mainly studied in the time [2]–[8] or frequency [9]–[11] domain, but control in the *spatial frequency* domain will be important in the future from the viewpoint of applications. An example is noise mitigation in sensor networks. When measuring the

temperature at different locations using a sensor network, as shown in Fig. 1, temperature generally has a low spatial frequency because its values remain similar at spatially close locations. In contrast, the noise has a high spatial frequency because its magnitude does not depend on sensor locations, as shown in Fig. 1. Hence, we can mitigate noise through spatial lowpass filtering. As illustrated by this example, there are applications in which we should use signal characteristics in the spatial frequency domain rather than those in the time or frequency domain.

This study employs graph signal processing [12]–[14] to introduce the concept of spatial frequencies to multi-agent control. Graph signal processing treats graph signals where the signal values are located on the vertices of graphs, and the relations between them are given by the edges of the graphs, as shown in Fig. 2. An example of a graph signal is

The associate editor coordinating the review of this manuscript and approving it for publication was Jianxiang Xi¹.

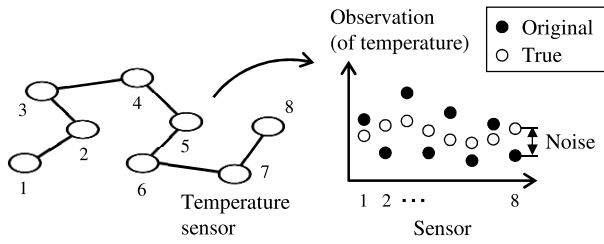


FIGURE 1. Sensor network for measuring temperature at different locations.

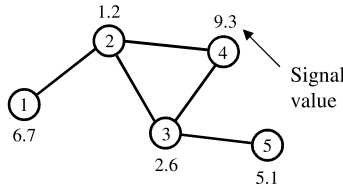


FIGURE 2. Graph signal comprising values (nodes) and their relation (edges).

the measurements from a sensor network, where the sensors and connections between them form the vertices and edges of a graph, respectively. The reason for employing graph signal processing is that it provides a spatial frequency-based representation of graph signals by the Fourier transform. We wish to use this technique to describe the behavior of systems based on spatial frequencies.

This study aims to establish a framework for the spatial frequency-based analysis and design of multi-agent systems and to verify its effectiveness and applicability. Our main contributions are threefold. First, we focus on the consensus [15] of multi-agent systems and analyze it from the viewpoint of spatial frequencies using the Fourier transform of graph signals. The consensus indicates that the states of all agents converge to an identical value, which is fundamental to multi-agent systems. We show that consensus by typical controllers can be considered as the spatial lowpass filtering of agent states, which demonstrates that a spatial frequency-based approach is available for multi-agent systems. Second, we present a controller design method in the spatial frequency domain. The proposed method provides controllers by which the feedback system has a desired spatial frequency characteristic known in advance. Further, we derive a sufficient condition for the spatial frequency characteristic under which the resulting controllers are distributed. Although we cannot always obtain distributed controllers by directly using the proposed method, the resulting controllers are ensured to be distributed if this condition is satisfied. Third, to demonstrate the effectiveness and applicability of the proposed design method, we present its application to distributed denoising in a sensor network. The results show that by considering spatial frequencies, the proposed method can effectively mitigate the measurement noise in the sensor network.

Before ending this section, we provide the following three remarks.

First, we comment on differences from related works.

Works on graph signal processing: Consensus by graph signal processing was studied in [16], [17]. They showed that the consensus is achieved by the lowpass filtering of a graph signal based on the relation between the initial and final agent states. In contrast, we show that consensus by typical controllers corresponds to the *iterative* lowpass filtering of a graph signal by focusing on the behavior of agents at each timestep. Meanwhile, [16], [17] focused only on achieving consensus and did not consider spatial frequencies, whereas we develop a method to design controllers in the spatial frequency domain and show its application to a sensor network. In addition, there are other related works [18], [19]. Shuman et al. [18] proposed a method to encode filters for graph signals into agents using Chebyshev polynomial approximation. However, this method is not directly applicable to our problem. As a typical setting in multi-agent control, we assume that each agent can only obtain information on the *differences* between its state and the states of neighboring agents and that the controllers of agents are memoryless, but such practical restrictions were not imposed in [18]. Segarra et al. [19] proposed a method to encode a filter for achieving a given linear transformation between two graph signals into agents. However, their work focused only on obtaining desired signal values given in advance, and thus, the results cannot be applied to spatial filtering where desired signal values are not explicitly provided.

Works on multi-agent control: Various studies on multi-agent control have been conducted to date. Recent studies include [20]–[24]. Zheng et al. [20] addressed a consensus problem for hybrid multi-agent systems composed of both agents with continuous-time dynamics and those with discrete-time dynamics. Wang and Ishii [21] considered resilient consensus through event-based communication for multi-agent systems with adversarial agents. Bowman et al. [22] proposed a self-triggered control method for cloud communication-based consensus. Besides consensus, Wang et al. [23] studied robust formation control for time-varying desired configurations in the presence of external disturbances. Liu et al. [24] also proposed a robust formation control method for a group of nonlinear and underactuated quadrotors subject to external disturbances. These studies, however, did not consider spatial frequencies for multi-agent systems.

Second, we would like to emphasize that this study does *not* aim to solve a consensus problem using graph signal processing. Our aim is to develop a framework for the analysis and design of multi-agent systems in the spatial frequency domain. As a result, the proposed controller design method is available not only for consensus but also for other tasks including distributed spatial filtering where the desired states of agents are generally different from each other, unlike in consensus.

Finally, this study is based on our preliminary results [25] presented at a conference, but provides the following novel contributions: (i) this study focuses on the spatial frequency-based analysis and design of multi-agent systems as an

extension of [25] where graph signal processing was only applied to multi-agent control; (ii) this paper provides complete explanations and rigorous proofs of the main results, omitted in [25]; (iii) we present a controller design method in the spatial frequency domain; (iv) to demonstrate the effectiveness and applicability of our design method, we show its application to distributed spatial filtering in a sensor network.

Notation: Throughout this paper, we use the following notation. Let \mathbb{R} and \mathbb{R}_+ denote the field of real numbers and the set of positive real numbers, respectively. We denote by I the identity matrix. For the numbers $x_1, x_2, \dots, x_n \in \mathbb{R}$, $\text{diag}(x_1, x_2, \dots, x_n)$ represents the diagonal matrix whose i -th entry on the diagonal is x_i ; we further let $[x_i]_{i \in \mathbb{I}} := [x_{i_1} \ x_{i_2} \ \dots \ x_{i_m}]^T \in \mathbb{R}^m$, where $\mathbb{I} := \{i_1, i_2, \dots, i_m\} \subseteq \{1, 2, \dots, n\}$. An example of the latter is $[x_i]_{i \in \mathbb{I}} := [x_1 \ x_3]^T$ for x_1, x_2, \dots, x_5 and $\mathbb{I} := \{1, 3\}$. We use $|\mathbb{S}|$ to represent the cardinality of the set \mathbb{S} . Moreover, for the graph Laplacian L of an undirected graph, the following properties [26] are employed.

- (L1) The matrix L is positive-semidefinite; that is, its eigenvalues are nonnegative real numbers. Moreover, all the eigenvalues are smaller than or equal to $2d_{\max}$ for the maximum degree $d_{\max} \in \mathbb{R}_+$ of the graph.
- (L2) If the graph is connected, L has only one zero eigenvalue.

II. GRAPH SIGNAL PROCESSING

In this section, an overview of graph signal processing is provided.

Consider the undirected graph $G = (\mathbb{V}, \mathbb{E})$ with n vertices, where $\mathbb{V} := \{1, 2, \dots, n\}$ is the vertex set and $\mathbb{E} \subset \mathbb{V} \times \mathbb{V}$ is the edge set. Then, a graph signal is defined as the pair of G and s , where $s \in \mathbb{R}^n$ is the collection of signal values located on all vertices. For example, we can represent the graph signal shown in Fig. 2 by G with $\mathbb{V} := \{1, 2, 3, 4, 5\}$ and $\mathbb{E} := \{(1, 2), (2, 3), (2, 4), (3, 4), (3, 5)\}$, and $s := [6.7 \ 1.2 \ 2.6 \ 9.3 \ 5.1]^T$.

The Fourier transform of graph signals is referred to as the graph Fourier transform, which is described below. Consider the graph signal (G, s) . Let $L \in \mathbb{R}^{n \times n}$ be the graph Laplacian of G and λ_i ($i \in \mathbb{V}$) be the eigenvalue of L whose modulus is the i -th smallest. Then, according to [13], [14], we define the graph Fourier transform $f \in \mathbb{R}^n$ as

$$f(\lambda_1, \lambda_2, \dots, \lambda_n) := V^T s, \tag{1}$$

where $V \in \mathbb{R}^{n \times n}$ is a matrix that satisfies $V^T = V^{-1}$ and

$$V^T L V = \Lambda \tag{2}$$

with $\Lambda := \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n)$. Note that the existence of such a V is guaranteed, because the matrix L is symmetric from property (L1) in Section I. The graph Fourier transform $f(\lambda_1, \lambda_2, \dots, \lambda_n)$ in (1) converts the graph signal (G, s) into one in the spatial frequency domain. More precisely,

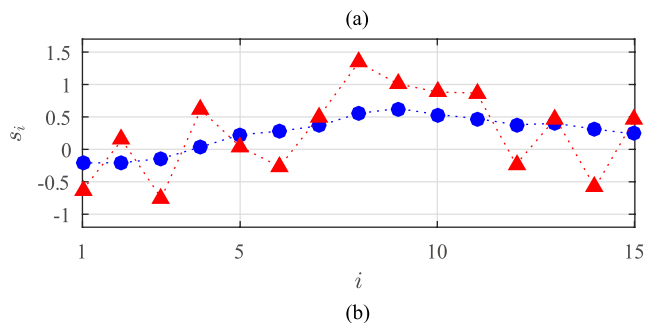
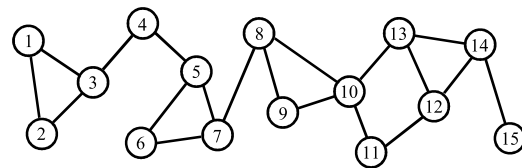


FIGURE 3. Two graph signals with (a) same graph but (b) different signal values at vertex i .

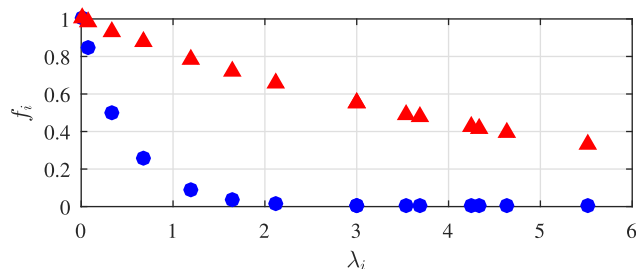


FIGURE 4. Graph Fourier transforms of graph signals shown in Fig. 3.

it represents the magnitude of the differences between signal values on the neighboring vertices of G . The eigenvalue λ_i ($i \in \mathbb{V}$) indicates the magnitude of the spatial frequency and the component of λ_i is given by f_i , i.e., the i -th entry of the vector f , where we note that $\lambda_1, \lambda_2, \dots, \lambda_n$ are nonnegative real numbers because of (L1). Meanwhile, focusing on the relation $V^T = V^{-1}$, we define the inverse graph Fourier transform as

$$s = V f(\lambda_1, \lambda_2, \dots, \lambda_n). \tag{3}$$

Let us illustrate the graph Fourier transform through an example. Consider the two graph signals depicted in Fig. 3, where s_i is the i -th entry of the vector s . We show the graph Fourier transforms of these signals in Fig. 4, where, similar to Fig. 3 (b), the triangles and circles are used to represent the two signals. Noting that the vertices whose indices are close to each other are connected on the graph in Fig. 3 (a), we can observe from Fig. 3 (b) that the spatial frequency of the graph signal indicated by the triangles is high compared to the other signal. By contrast, we can observe from Fig. 4 that this graph signal has large components in the high spatial frequency range compared to the other signal. Hence, it is demonstrated that a spatial frequency-based representation of graph signals can be obtained by the graph Fourier transform.

For extracting specific frequency components from graph signals, such as denoising, we use the filtering in the spatial frequency domain. A filter is given by

$$\tilde{s} = V \text{diag}(h(\lambda_1), h(\lambda_2), \dots, h(\lambda_n)) V^T s, \quad (4)$$

where $\tilde{s} \in \mathbb{R}^n$ is the vector containing the signal values of a filtered graph signal and $h : \mathbb{R}_+ \cup \{0\} \rightarrow \mathbb{R}$ is a filter function. As an example of the filter function, if h is chosen such that $|h(\lambda_i)|$ decreases as i increases, a lowpass filter is obtained because higher frequency components decrease. Equations (1), (3), and (4) imply that spatial frequency domain filtering comprises the following three steps: (i) obtaining the graph Fourier transform f of a given graph signal, (ii) multiplying the resulting f by the matrix $\text{diag}(h(\lambda_1), h(\lambda_2), \dots, h(\lambda_n))$ given by the filter function, and (iii) performing the inverse graph Fourier transform.

III. ANALYSIS OF MULTI-AGENT SYSTEMS IN SPATIAL FREQUENCY DOMAIN: CASE OF CONSENSUS

In this section, we focus on consensus, which is fundamental to multi-agent systems, and analyze it from the viewpoint of spatial frequencies using graph signal processing techniques.

A. MULTI-AGENT CONSENSUS

Consider the multi-agent system Σ with n agents, illustrated in Fig. 5. Agent i ($i \in \{1, 2, \dots, n\}$) is modeled as

$$x_i(t + 1) = x_i(t) + u_i(t), \quad (5)$$

where $x_i(t) \in \mathbb{R}$ and $u_i(t) \in \mathbb{R}$ are the agent state and the control input, respectively. We suppose that agent i can obtain information on the differences between its own state and the states of neighboring agents over the network, i.e., $[x_j(t) - x_i(t)]_{j \in \mathbb{N}_i} \in \mathbb{R}^{|\mathbb{N}_i|}$ where $\mathbb{N}_i \subset \{1, 2, \dots, n\}$ is the index set of the neighbors (see Section I for the definition of $[x_j(t) - x_i(t)]_{j \in \mathbb{N}_i}$). The network is modeled as the undirected graph $G = (\mathbb{V}, \mathbb{E})$ where the vertex set \mathbb{V} represents the agents and the edge set $\mathbb{E} \subset \mathbb{V} \times \mathbb{V}$ represents the connections between them. Moreover, $x(t) \in \mathbb{R}^n$ denotes all the agent states, that is, $x(t) := [x_1(t) \ x_2(t) \ \dots \ x_n(t)]^T$.

Consensus is defined as the convergence of every agent state to an identical value:

$$\lim_{t \rightarrow \infty} (x_i(t) - x_j(t)) = 0 \quad \forall (i, j) \in \mathbb{V} \times \mathbb{V}. \quad (6)$$

Consider the following local controller K_i for agent i ($i \in \mathbb{V}$):

$$K_i : u_i(t) = k \sum_{j \in \mathbb{N}_i} (x_j(t) - x_i(t)), \quad (7)$$

where $k \in \mathbb{R}_+$ is the gain of the controller. Then, according to [26], if the following two conditions hold, (6) holds for every $x(0) \in \mathbb{R}^n$.

- (C1) The graph G is connected.
- (C2) For the maximum degree $d_{\max} \in \mathbb{R}_+$ of G , the gain k satisfies $k < 1/d_{\max}$.

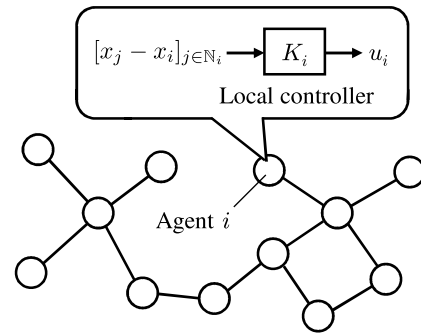


FIGURE 5. Multi-agent system Σ .

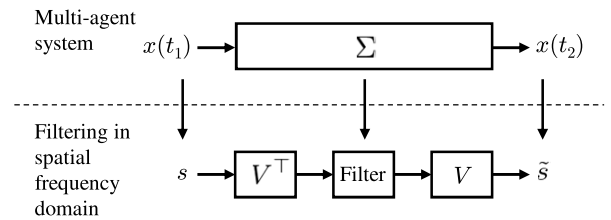


FIGURE 6. Analysis in spatial frequency domain.

B. ANALYSIS IN SPATIAL FREQUENCY DOMAIN

From the discussion in Section II, we notice that the pair of the network topology G and the collective state $x(t)$ can be considered as a graph signal. This is because G and $x(t)$ can be considered as the graph that specifies the signal structure and the collective signal value, respectively. Based on this observation, we analyze consensus in the spatial frequency domain. The concept of the analysis is illustrated in Fig. 6. For timesteps t_1 and t_2 satisfying $t_1 < t_2$, we regard $x(t_1)$ and $x(t_2)$ as s and \tilde{s} in the spatial frequency domain filtering (4), respectively. Then, we can consider that there is a relation between the multi-agent system and the filtering (4) if there exists a filter (i.e., a function h) such that the transformation from $x(t_1)$ to $x(t_2)$ is equivalent to (4). By investigating this filter, we can characterize consensus in the spatial frequency domain.

Using (5) and (7), we can write the feedback system in vector form:

$$x(t + 1) = (I - kL)x(t). \quad (8)$$

Applying $V^T = V^{-1}$ and (2) to (8) yields

$$\begin{aligned} x(t + 1) &= (VV^T - kV\Lambda V^T)x(t) \\ &= V(I - k\Lambda)V^T x(t) \\ &= V \text{diag}(1 - k\lambda_1, 1 - k\lambda_2, \dots, 1 - k\lambda_n) V^T x(t). \end{aligned} \quad (9)$$

In (9), we regard $x(t)$, $x(t + 1)$, and $1 - k\lambda_i$ ($i \in \mathbb{V}$) as s , \tilde{s} , and $h(\lambda_i)$ ($i \in \mathbb{V}$) in (4), respectively, which yields the following theorem.

Theorem 1: Consider the multi-agent system Σ with the local controllers K_1, K_2, \dots, K_n given by (7). For the frequency variable $\lambda \in \mathbb{R}_+ \cup \{0\}$, let the filter function be of

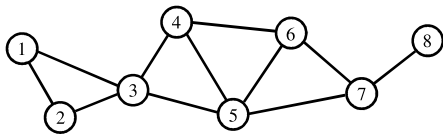


FIGURE 7. Network topology G.

the form $h(\lambda) := 1 - k\lambda$. Then, the behavior of the system Σ is equivalent to filtering the graph signal $(G, x(t))$ using $h(\lambda)$ at each timestep $t \in \{0, 1, \dots\}$.

In this theorem, $h(\lambda) := 1 - k\lambda$ provides a filter such that nonzero frequency components decrease under condition (C2) discussed in Section III-A. The proof is given as follows. The filter function, $h(\lambda) := 1 - k\lambda$, implies

$$\begin{aligned} 1 - k\lambda &\geq 1 - k\lambda_n \\ &\geq 1 - 2kd_{\max} \\ &> 1 - 2 \\ &= -1 \end{aligned}$$

for every $\lambda \in [0, \lambda_n]$, where $k \in \mathbb{R}_+$, (L1), and (C2) are used to derive the first, second, and third relations, respectively. Meanwhile, $1 > 1 - k\lambda$ holds for every $\lambda \in (0, \lambda_n]$ from $k \in \mathbb{R}_+$. Hence, we obtain $|1 - k\lambda| < 1$ for every $\lambda \in (0, \lambda_n]$, and the proof is complete. Regarding this as a lowpass filter characteristic and considering Theorem 1, we can conclude that consensus obtained from (7) corresponds to the lowpass filtering of the graph signal $(G, x(t))$, i.e., the spatial lowpass filtering of the agent states, at every timestep $t \in \{0, 1, \dots\}$.

As an example, consider consensus for $n := 8$ and the network topology G shown in Fig. 7. Let the local controllers K_1, K_2, \dots, K_8 be given by (7) and $k := 0.05$. In this setting, conditions (C1) and (C2) are satisfied. Fig. 8 shows the time evolutions of the agent states $x_1(t), x_2(t), \dots, x_8(t)$ for $x(0) := [-9 \ 6 \ -7 \ 8 \ -3 \ 9 \ -8 \ 4]^T$, where consensus is achieved. Fig. 9 shows each agent state at the timesteps $t = 0, 8$, and 80. We observe that high spatial frequency components decrease with time.

Remark 1: The difficulty of the spatial frequency-based analysis considered here concerns how the concept of spatial frequencies should be introduced to the multi-agent system Σ . We overcome this difficulty by using the idea of graph signal processing and associating the system Σ with the filtering (4) in the spatial frequency domain as shown in Fig. 6.

Remark 2: The results presented here not only provide the characterization of consensus in the spatial frequency domain but also imply that controllers of the form (7) are important in the spatial frequency-based design. As shown in Fig. 6, V^T and V appear in the spatial frequency domain filtering; thus, all controllers do not provide feedback systems that can be handled in our framework. However, as demonstrated in this section, controllers of the form (7) provide feedback systems that can be handled in the spatial frequency domain. The results in the next section are based on this fact.

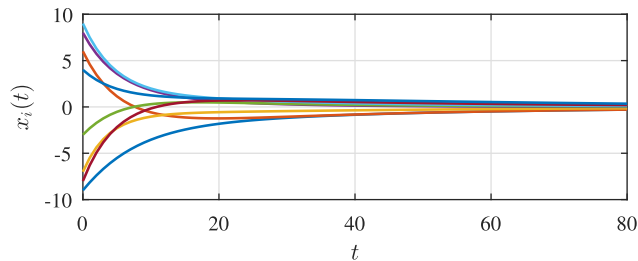


FIGURE 8. Time evolutions of agent states by (7).

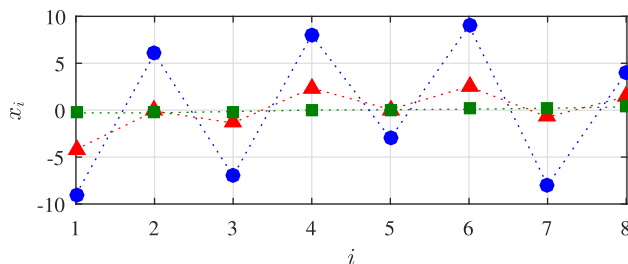


FIGURE 9. Each agent state at $t = 0$ (circle), 8 (triangle), and 80 (square).

IV. CONTROLLER DESIGN IN SPATIAL FREQUENCY DOMAIN

Next, taking consensus as an example, we present a controller design method in the spatial frequency domain. The effectiveness and applicability of our design method are demonstrated through an example of distributed spatial filtering in a sensor network.

A. PROBLEM FORMULATION

We again consider the multi-agent system Σ discussed in Section III. Let the local controller K_i ($i \in \mathbb{V}$) be given by

$$K_i : u_i(t) = g([x_j(t) - x_i(t)]_{j \in \mathbb{N}_i}, t), \quad (10)$$

where $[x_j(t) - x_i(t)]_{j \in \mathbb{N}_i}$, $u_i(t)$, and $g : \mathbb{R}^{|\mathbb{N}_i|} \times \{0, 1, \dots\} \rightarrow \mathbb{R}$ are the input, the output, and a function specifying the controller, respectively.

Then, we address the following problem.

Problem 1: For the multi-agent system Σ , design the local controllers K_1, K_2, \dots, K_n (i.e., the function g) that satisfy (6) for every $x(0) \in \mathbb{R}^n$.

B. PROPOSED METHOD

To solve Problem 1 in the spatial frequency domain, we recall Fig. 6. We suppose here $t_1 := 0$ and $t_2 := \infty$; that is, the initial state $x(0)$ and the final state $x(\infty)$ correspond to s and \tilde{s} , respectively. Then, the multi-agent system Σ works as a spatial filter for the graph signal $(G, x(0))$. Therefore, we can achieve (6) by designing the local controllers K_1, K_2, \dots, K_n such that the resulting system Σ works as a filter for achieving consensus.

Based on this idea, we propose the following method to design the local controllers K_1, K_2, \dots, K_n .

- (i) Select a filter (i.e., a function h) for achieving consensus as the final state.

- (ii) Encode the selected filter into K_1, K_2, \dots, K_n such that the relation between the initial and final states of the resulting system Σ is equivalent to (4).

However, we cannot always encode a given filter into the local controllers K_1, K_2, \dots, K_n because only restricted information (i.e., $[x_j(t) - x_i(t)]_{j \in \mathbb{N}_i}$) is available to each local controller, as shown in (10). How do we select the filter to obtain the distributed controllers?

An answer is given by the following result.

Theorem 2: Consider the multi-agent system Σ . Let the filter function be given by

$$h(\lambda) := a_m \lambda^m + a_{m-1} \lambda^{m-1} + \dots + a_1 \lambda + 1, \quad (11)$$

where $a_1, a_2, \dots, a_m \in \mathbb{R}$ are the coefficients of the polynomial with the degree $m \in \mathbb{R}_+$, and we assume that the roots r_1, r_2, \dots, r_m are real numbers. Further, let K_1, K_2, \dots, K_n be given by (10) and

$$g([x_j(t) - x_i(t)]_{j \in \mathbb{N}_i}, t) := k(t) \sum_{j \in \mathbb{N}_i} (x_j(t) - x_i(t)) \quad (12)$$

with

$$k(t) := \begin{cases} \frac{1}{r_{t+1}} & \text{if } t \in \{0, 1, \dots, m-1\}, \\ 0 & \text{otherwise.} \end{cases} \quad (13)$$

Then, the behavior of the system Σ is equivalent to filtering the graph signal $(G, x(0))$ using $h(\lambda)$.

Proof: It follows from (5), (10), (12), and (13) that

$$x(m) = V \text{diag}(h(\lambda_1), h(\lambda_2), \dots, h(\lambda_n)) V^\top x(0). \quad (14)$$

The proof is given in Appendix A. Using (10), (12), and (13), we can show that $u_i(t) = 0$ holds for every $i \in \mathbb{V}$ and $t \in \{m, m+1, \dots\}$, which yields $x(m) = x(\infty)$ because of (5). By applying this to (14) and considering $x(\infty)$ and $x(0)$ as \tilde{s} and s in (4), respectively, we complete the proof. ■

It is remarked in (13) that the index of the root r_{t+1} depends on the timestep t .

From Theorem 2, we see that (10), (12), and (13) construct the local controller K_i ($i \in \mathbb{V}$) when the filter function h is selected as the polynomial given by (11). Thus, by selecting the filter based on (11), we can obtain distributed controllers.

Based on the above discussion, we propose Algorithm 1 as a method to solve Problem 1 in the spatial frequency domain. In Step 1, we should select the filter function h such that $|h(\lambda_1)| = 1$ (i.e., $|h(0)| = 1$) and $|h(\lambda_i)|$ is sufficiently small for $i \in \{2, 3, \dots, n\}$. The reason is as follows. As explained in Section III-B, consensus is achieved by attenuating the components of frequencies other than zero. Moreover, when the graph G is connected, L has only one zero eigenvalue, i.e., $\lambda_1 = 0$ and $\lambda_2 > 0$, as mentioned in (L2). Hence, we can achieve consensus by selecting an h such that the frequency components at $\lambda_2, \lambda_3, \dots, \lambda_n$ decrease.

Remark 3: The difficulty in obtaining Theorem 2 concerns how the controller of each individual agent should be found such that the resulting network can be analyzed in a manner similar to that discussed in Section III-B. We overcome this

Algorithm 1 Design of Consensus Controllers in Spatial Frequency Domain

Step 1 Select the filter function h as a lowpass filter for achieving consensus.

Step 2 Describe h as the polynomial given by (11) with real roots through, for example, fitting.

Step 3 Construct the local controllers K_1, K_2, \dots, K_n using (10), (12), (13), and the roots of the resulting h .

difficulty by focusing on the fact that one of such controllers is of the form (7) as stated in Remark 2 and by employing it with a time-varying gain.

Remark 4: Although a similar condition for the filter function h can be found in [16], [17], our main contribution here is to present a *generalized* method to design the local controllers K_1, K_2, \dots, K_n for achieving a desired spatial frequency characteristic. Compared to the methods in [16], [17], our method can handle not only the consensus problem but also other problems by changing h in Step 1 of Algorithm 1. This is demonstrated in Section IV-D.

C. ILLUSTRATIVE EXAMPLE

Let us illustrate the proposed method through an example. Consider consensus for $n := 8$ and the network topology G shown in Fig. 7. The desired filter function is selected as $h(\lambda) := e^{-5\lambda}$. This and $\lambda_1, \lambda_2, \dots, \lambda_8$ are depicted as the thick line and circles in Fig. 10, respectively, where it should be noted that a few of $\lambda_1, \lambda_2, \dots, \lambda_8$ are equal. It turns out that the filter function satisfies the condition described in Section IV-B. Then, the selected $h(\lambda)$ is approximated by a polynomial written as (11) with real roots, which gives

$$h(\lambda) := -0.00122\lambda^7 + 0.0283\lambda^6 - 0.265\lambda^5 + 1.29\lambda^4 - 3.49\lambda^3 + 5.11\lambda^2 - 3.69\lambda + 1 \quad (15)$$

with $m := 7$. The thin line in Fig. 10 shows $h(\lambda)$ in (15). It turns out that the approximation captures the selected filter function well. For the approximated $h(\lambda)$, we calculate its roots as $r_1 = 5.91, r_2 = 5.36, r_3 = 4.47, r_4 = 3.37, r_5 = 2.22, r_6 = 1.17$, and $r_7 = 0.663$. Hence, we employ the local controllers K_1, K_2, \dots, K_8 determined by (10), (12), (13), and r_1, r_2, \dots, r_7 . For $x(0) := [6 \ -2 \ -5 \ 0 \ 3 \ 8 \ -9 \ 1]^\top$, the time evolutions of the agent states $x_1(t), x_2(t), \dots, x_8(t)$ are illustrated in Fig. 11, where consensus is achieved using the proposed method.

Remark 5: Even though we must only achieve consensus asymptotically in Problem 1 from (6), if the proposed method achieves consensus, it is completed in a finite time, i.e., m timesteps. This is proven as follows. As $u_i(t) = 0$ holds for every $i \in \mathbb{V}$ and $t \in \{m, m+1, \dots\}$ (see the proof of Theorem 2), it follows from (5) that $x(t)$ does not change for $t \geq m$. Thus, the control of $x(t)$ by the proposed method finishes at $t = m$, which proves the statement. This statement can be interpreted as follows. From (23) in Appendix A, the filter function in (11) can be written as the product of m

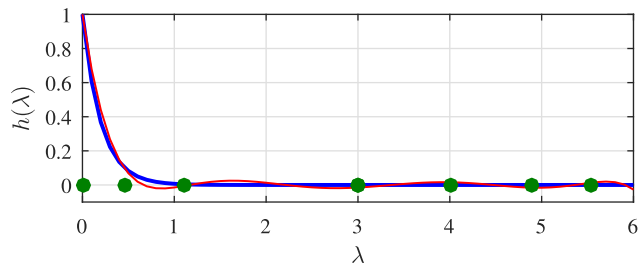


FIGURE 10. Filter function $h(\lambda) := e^{-5\lambda}$ (thick line) and its polynomial approximation (thin line), where the circles represent $\lambda_1, \lambda_2, \dots, \lambda_8$.

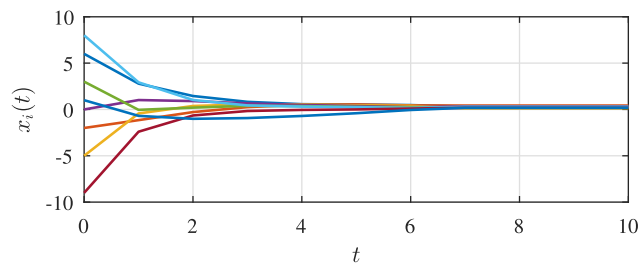


FIGURE 11. Time evolutions of agent states by (10), (12), and (13).

first-order polynomials. Each polynomial corresponds to the transition from $x(t)$ to $x(t + 1)$ following the discussion in Section III-B. Therefore, the proposed controllers obtained from the filter function in (11) achieve state transition in m timesteps.

Remark 6: We comment on the choice of the degree m of the polynomial in (11). The error in the approximation of the filter function in Step 2 of Algorithm 1 decreases as m increases, which gives better controllers in terms of achieving the desired filtering performance. However, Remark 5 implies that the agent states have slower convergence as m increases. In addition, Theorem 2 requires the roots of the polynomial in (11) to be real, but it is generally difficult to find a high-order polynomial without complex roots. From these facts, we should determine m by considering the trade-off between filtering performance and convergence speed, as long as the resulting polynomial does not have complex roots.

Remark 7: The proposed method guarantees the scalability of the entire system in the following two points. First, from (10), (12), and (13), the structure and gain of the resulting local controller K_i are the same for all $i \in \mathbb{V}$. Second, the filtering of graph signals by directly using (4) requires the eigendecomposition of the graph Laplacian L , whose computational complexity increases with n , but the proposed method does not necessarily require it. As demonstrated in the example above, we can obtain K_1, K_2, \dots, K_n by simply calculating the roots of the polynomial in (11) and using (10), (12), and (13). Meanwhile, if the eigenvalues $\lambda_1, \lambda_2, \dots, \lambda_n$ are obtained, the time and effort spent to find an appropriate filter function h are reduced because they correspond to the spatial frequencies at which target signals have components.

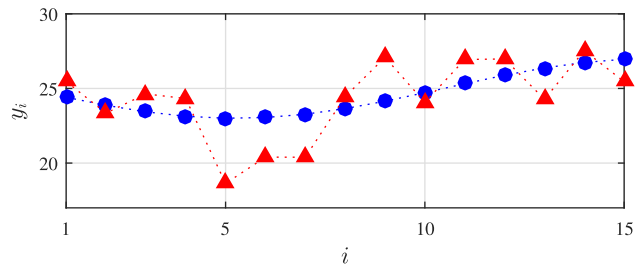


FIGURE 12. Noisy observation from each sensor (triangle) and its true value (circle).

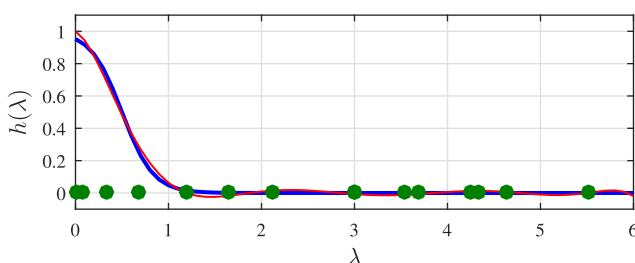


FIGURE 13. Filter function $h(\lambda) := -1/(1 + e^{-6(\lambda-0.5)}) + 1$ (thick line) and its polynomial approximation (thin line), where the circles represent $\lambda_1, \lambda_2, \dots, \lambda_{15}$.

D. APPLICATION: DISTRIBUTED SPATIAL FILTERING IN SENSOR NETWORK

As previously mentioned, the proposed method allows us to design the local controllers K_1, K_2, \dots, K_n for achieving a desired spatial frequency characteristic, and can be applied to other tasks as well as consensus. To demonstrate this, we illustrate distributed spatial filtering in a sensor network.

Consider the multi-agent system Σ with $n := 15$ and the network topology G shown in Fig. 3 (a), where the agents correspond to sensors. Fig. 12 depicts a noisy observation $y_i \in \mathbb{R}$ from sensor i ($i \in \mathbb{V}$) and the true value of the observation. Then, we assume that each sensor i estimates the true value of y_i in a distributed manner and design distributed denoising filters using the proposed method.

Considering that noise has a high spatial frequency compared to the true values, we select the filter function $h(\lambda) := -1/(1 + e^{-6(\lambda-0.5)}) + 1$ for mitigating noise. Fig. 13 depicts $h(\lambda)$ in the same manner as Fig. 10, where we note that a few of $\lambda_1, \lambda_2, \dots, \lambda_{15}$ are equal. It turns out that the filter has a lowpass characteristic, where we select a different filter from that shown in Fig. 10 for preserving the low-frequency components. Then, the selected $h(\lambda)$ is approximated by a polynomial written as (11) with real roots, which gives

$$h(\lambda) := -0.000587\lambda^8 + 0.0150\lambda^7 - 0.157\lambda^6 + 0.869\lambda^5 - 2.68\lambda^4 + 4.44\lambda^3 - 3.13\lambda^2 - 0.293\lambda + 1 \quad (16)$$

with $m := 8$. The thin line in Fig. 13 shows $h(\lambda)$ in (16). We observe that the approximation captures the selected filter function well. The roots of the approximated $h(\lambda)$ are $r_1 = 5.94, r_2 = 5.50, r_3 = 4.78, r_4 = 3.86, r_5 = 2.85, r_6 = 1.87, r_7 = 1.19,$ and $r_8 = -0.445$. Hence, the distributed

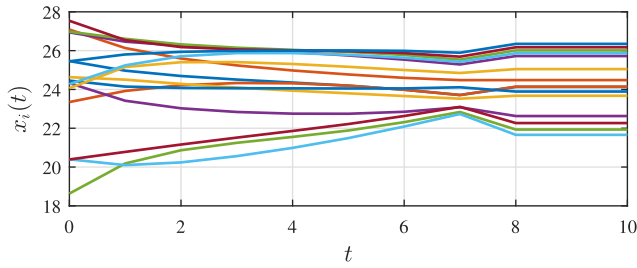


FIGURE 14. Time evolutions of agent states by proposed denoising filters.

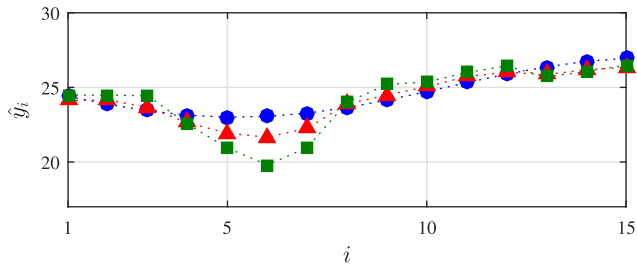


FIGURE 15. Estimation results obtained using proposed method (triangles) and average filter (squares), and true values (circles).

filters are obtained as the local controllers K_1, K_2, \dots, K_{15} determined by (10), (12), (13), and r_1, r_2, \dots, r_8 . That is, our filters correspond to the consensus controllers discussed in Section IV-C with a different gain $k(t)$. These filters are applied to $x(0) := y$ with $y := [y_1 \ y_2 \ \dots \ y_{15}]^T$.

As the filtering process, the time evolutions of the states $x_1(t), x_2(t), \dots, x_{15}(t)$ are shown in Fig. 14. We observe that all the states do not converge to an identical value unlike in consensus because the filter function (*i.e.*, the gain $k(t)$) is designed for achieving not consensus but noise mitigation. The resulting estimate $\hat{y}_i := x_i(10)$ ($i \in \mathbb{V}$) and the true values are shown in Fig. 15. We see that the estimates agree well with the true values. This is because we considered the difference between the spatial frequencies of the true values and noise and designed an appropriate filter for mitigating noise. Fig. 15 also shows the estimates obtained using an average filter [27] which is typically used for spatial lowpass filtering in image processing. The average filter provides the estimate \hat{y}_i ($i \in \mathbb{V}$) as

$$\hat{y}_i := \frac{y_i + \sum_{j \in \mathbb{N}_i} y_j}{|\mathbb{N}_i| + 1}. \quad (17)$$

For the vector y^* consisting of the true values and $\hat{y} := [\hat{y}_1 \ \hat{y}_2 \ \dots \ \hat{y}_{15}]^T$, the estimation error $\|y^* - \hat{y}\|_\infty$ is 1.39 for the proposed method and 3.24 for the average filter. Therefore, we can conclude that the filter designed using the proposed method provides high accuracy compared to the average filter.

V. CONCLUSION

In this study, we have addressed the spatial frequency-based analysis and design of multi-agent systems and shown the usefulness for distributed spatial filtering in sensor networks.

First, by focusing on the graph Fourier transform, we have analyzed the consensus in the spatial frequency domain. Then, we have shown that consensus by typical controllers is equivalent to the spatial lowpass filtering of agent states in the sense of decreasing nonzero frequency components. This demonstrates that the spatial frequency-based approach can be used for multi-agent systems. Second, a controller design method in the spatial frequency domain has been presented, where a desired spatial filter is realized as the feedback system. We have proven that this method gives distributed controllers if the filter function is selected as a polynomial. Finally, to demonstrate the effectiveness and applicability of our design method, we have presented its application to distributed noise mitigation in a sensor network.

A limitation of the design method presented in this paper is that it can handle only specific types of agents and filters. For example, the dynamics of each agent is restricted to single integrator dynamics. Relaxing this constraint is part of our future work.

APPENDIX A PROOF OF (14)

Using $x(0)$, we can express $x(m)$ as

$$\begin{aligned} x(m) &= \left[\prod_{j=0}^{m-1} \left(I - \frac{1}{r_{m-j}} L \right) \right] x(0) \\ &= \left[\prod_{j=0}^{m-1} \left(V V^T - \frac{1}{r_{m-j}} V \Lambda V^T \right) \right] x(0) \\ &= \left[\prod_{j=0}^{m-1} V \left(I - \frac{1}{r_{m-j}} \Lambda \right) V^T \right] x(0) \\ &= V \left[\prod_{j=0}^{m-1} \left(I - \frac{1}{r_{m-j}} \Lambda \right) \right] V^T x(0), \quad (18) \end{aligned}$$

where the first equality is derived from (5), (10), (12), and (13), the second one follows from $V^T = V^{-1}$ and (2), the third one is trivial, and the final one is obtained using $V^T V = I$. Noting that $I - (1/r_{m-j})\Lambda$ is a diagonal matrix for every $j \in \{0, 1, \dots, m-1\}$, we obtain

$$\begin{aligned} x(m) &= V \text{diag} \left(\prod_{j=0}^{m-1} \left(1 - \frac{\lambda_1}{r_{m-j}} \right), \prod_{j=0}^{m-1} \left(1 - \frac{\lambda_2}{r_{m-j}} \right), \right. \\ &\quad \left. \dots, \prod_{j=0}^{m-1} \left(1 - \frac{\lambda_n}{r_{m-j}} \right) \right) V^T x(0). \quad (19) \end{aligned}$$

Next, we prove that

$$h(\lambda_i) = \prod_{j=0}^{m-1} \left(1 - \frac{\lambda_i}{r_{m-j}} \right) \quad (20)$$

for every $i \in \mathbb{V}$. Equation (11) is factorized as

$$h(\lambda) = a_m \prod_{j=0}^{m-1} (\lambda - r_{m-j}). \quad (21)$$

By a simple calculation, it follows that

$$\begin{aligned} & a_m \prod_{j=0}^{m-1} (\lambda - r_{m-j}) \\ &= a_m \prod_{j=0}^{m-1} \left(r_{m-j} \left(\frac{\lambda}{r_{m-j}} - 1 \right) \right) \\ &= a_m \left[\prod_{j=0}^{m-1} r_{m-j} \right] \left[\prod_{j=0}^{m-1} \left(\frac{\lambda}{r_{m-j}} - 1 \right) \right] \\ &= (-1)^m a_m \left[\prod_{j=0}^{m-1} r_{m-j} \right] \left[\prod_{j=0}^{m-1} \left(1 - \frac{\lambda}{r_{m-j}} \right) \right]. \quad (22) \end{aligned}$$

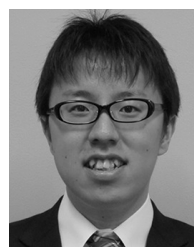
From the fact that the constant term of (11) is 1, we obtain $(-1)^m a_m \prod_{j=0}^{m-1} r_{m-j} = 1$. Therefore,

$$h(\lambda) = \left[\prod_{j=0}^{m-1} \left(1 - \frac{\lambda}{r_{m-j}} \right) \right] \quad (23)$$

holds. Substituting $\lambda = \lambda_i$ ($i \in \mathbb{V}$) for (23) results in (20) for every $i \in \mathbb{V}$. This, together with (19), proves (14).

REFERENCES

- [1] M. Mesbahi and M. Egerstedt, *Graph Theoretic Methods in Multiagent Networks*. Princeton, NJ, USA: Princeton Univ. Press, 2010.
- [2] J. A. Fax and R. M. Murray, "Information flow and cooperative control of vehicle formations," *IEEE Trans. Autom. Control*, vol. 49, no. 9, pp. 1465–1476, Sep. 2004.
- [3] J. Cortés, S. Martínez, and F. Bullo, "Spatially-distributed coverage optimization and control with limited-range interactions," *ESAIM, Control, Optim. Calc. Var.*, vol. 11, no. 4, pp. 691–719, Oct. 2005.
- [4] H. G. Tanner, A. Jadbabaie, and G. J. Pappas, "Flocking in fixed and switching networks," *IEEE Trans. Autom. Control*, vol. 52, no. 5, pp. 863–868, May 2007.
- [5] J. R. Marden, G. Arslan, and J. S. Shamma, "Cooperative control and potential games," *IEEE Trans. Syst. Man, Cybern. B, Cybern.*, vol. 39, no. 6, pp. 1393–1407, Dec. 2009.
- [6] S. Izumi, S.-I. Azuma, and T. Sugie, "Distributed hybrid controllers for multi-agent mass games by a variable number of player agents," *Asian J. Control*, vol. 17, no. 3, pp. 762–774, May 2015.
- [7] K. Sakurama, S.-I. Azuma, and T. Sugie, "Distributed controllers for multi-agent coordination via gradient-flow approach," *IEEE Trans. Autom. Control*, vol. 60, no. 6, pp. 1471–1485, Jun. 2015.
- [8] S. Izumi, S.-I. Azuma, and T. Sugie, "Stabilization by controller networks," *Syst. Control Lett.*, vol. 94, pp. 77–83, Aug. 2016.
- [9] R. Olfati-Saber and J. S. Shamma, "Consensus filters for sensor networks and distributed sensor fusion," in *Proc. 44th IEEE Conf. Decis. Control (CDC) Eur. Control Conf. (ECC)*, Dec. 2005, pp. 6698–6703.
- [10] D. Lee and M. W. Spong, "Agreement with non-uniform information delays," in *Proc. Amer. Control Conf. (ACC)*, Jun. 2006, pp. 756–761.
- [11] Y.-P. Tian and C.-L. Liu, "Consensus of multi-agent systems with diverse input and communication delays," *IEEE Trans. Autom. Control*, vol. 53, no. 9, pp. 2122–2128, Oct. 2008.
- [12] D. K. Hammond, P. Vandergheynst, and R. Gribonval, "Wavelets on graphs via spectral graph theory," *Appl. Comput. Harmon. Anal.*, vol. 30, no. 2, pp. 129–150, Mar. 2011.
- [13] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Process. Mag.*, vol. 30, no. 3, pp. 83–98, May 2013.
- [14] A. Sandryhaila and J. M. F. Moura, "Discrete signal processing on graphs: Frequency analysis," *IEEE Trans. Signal Process.*, vol. 62, no. 12, pp. 3042–3054, Jun. 2014.
- [15] W. Ren and R. W. Beard, *Distributed Consensus in Multi-vehicle Cooperative Control*, London, U.K.: Springer, 2008.
- [16] A. Sandryhaila, S. Kar, and J. M. F. Moura, "Finite-time distributed consensus through graph filters," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2014, pp. 1080–1084.
- [17] J.-W. Yi and L. Chai, "Graph filter design for multi-agent system consensus," in *Proc. 56th IEEE Conf. Decis. Control (CDC)*, Dec. 2017, pp. 1082–1087.
- [18] D. I. Shuman, P. Vandergheynst, and P. Frossard, "Chebyshev polynomial approximation for distributed signal processing," in *Proc. Int. Conf. Distrib. Comput. Sensor Syst. Workshops (DCOSS)*, Jun. 2011, pp. 1–8.
- [19] S. Segarra, A. G. Marques, and A. Ribeiro, "Optimal graph-filter design and applications to distributed linear network operators," *IEEE Trans. Signal Process.*, vol. 65, no. 15, pp. 4117–4131, Aug. 2017.
- [20] Y. Zheng, J. Ma, and L. Wang, "Consensus of hybrid multi-agent systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 4, pp. 1359–1365, Apr. 2018.
- [21] Y. Wang and H. Ishii, "Resilient consensus through event-based communication," *IEEE Trans. Control Netw. Syst.*, to be published, doi: 10.1109/TCNS.2019.2924235.
- [22] S. L. Bowman, C. Nowzari, and G. J. Pappas, "Consensus of multi-agent systems via asynchronous cloud communication," *IEEE Control Netw. Syst.*, to be published, doi: 10.1109/TCNS.2019.2935626.
- [23] L. Wang, J. Xi, M. He, and G. Liu, "Robust time-varying formation design for multi-agent systems with disturbances: Extended-state-observer method," Sep. 2019, *arXiv:1909.08974*. [Online]. Available: <http://arxiv.org/abs/1909.08974>
- [24] H. Liu, T. Ma, F. L. Lewis, and Y. Wan, "Robust formation control for multiple quadrotors with nonlinearities and disturbances," *IEEE Trans. Cybern.*, to be published, doi: 10.1109/TCYB.2018.2875559.
- [25] S. Izumi, S.-I. Azuma, and T. Sugie, "On a relation between graph signal processing and multi-agent consensus," in *Proc. 55th IEEE Conf. Decis. Control (CDC)*, Dec. 2016, pp. 957–961.
- [26] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proc. IEEE*, vol. 95, no. 1, pp. 215–233, Jan. 2007.
- [27] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 2nd ed. Upper Saddle River, NJ, USA: Prentice-Hall, 2001.



SHINSAKU IZUMI (Member, IEEE) received the M.S. and Ph.D. degrees in informatics from Kyoto University, Japan, in 2012 and 2015, respectively. He is currently an Assistant Professor with the Faculty of Computer Science and Systems Engineering, Okayama Prefectural University, Japan. His research interests include networked control systems and multiagent systems.



SHUN-ICHI AZUMA (Senior Member, IEEE) was born in Tokyo, Japan, in 1976. He received the B.Eng. degree in electrical engineering from Hiroshima University, Higashi Hiroshima, Japan, in 1999, and the M.Eng. and Ph.D. degrees in control engineering from Tokyo Institute of Technology, Tokyo, Japan, in 2001 and 2004, respectively. Prior to joining Nagoya University, he was an Assistant Professor and an Associate Professor with the Department of Systems Science, Graduate School of Informatics, Kyoto University, Uji, Japan, from 2005 to 2011 and from 2011 to 2017, respectively. He is currently a Professor with the Department of Mechanical Systems Engineering, Graduate School of Engineering, Nagoya University, Nagoya, Japan. His research interests include hybrid systems and networked systems. He serves as an Associate Editor for the IEEE TRANSACTIONS ON CONTROL OF NETWORK SYSTEMS, since 2013, the *IFAC Journal Automatica*, since 2014, the IEEE CONTROL SYSTEMS LETTERS, since 2016, and the *Nonlinear Analysis: Hybrid Systems*, since 2017.



TOSHIHARU SUGIE (Fellow, IEEE) received the B.E., M.E., and Ph.D. degrees in engineering from Kyoto University, Japan, in 1976, 1978, and 1985, respectively. From 1978 to 1980, he was a Research Member of the Musashino Electric Communication Laboratory, NTT, Musashino, Japan. From 1984 to 1988, he was a Research Associate of the Department of Mechanical Engineering, University of Osaka Prefecture, Osaka. From 1988 to 2019, he worked at Kyoto University, where he was a Professor of the Department of Systems Science, since 1997. In 2019, he joined Osaka University as a Research Member of the Komatsu MIRAI Construction Equipment Cooperative Research Center, Graduate School of Engineering. His research interests are in robust control, identification for control, and control application to mechanical systems. He has served as an Editor of *Automatica*, and was also an Associate Editor of the *Asian Journal of Control* and the *International Journal of Systems Science*.

• • •