Edge Agreement: Graph-Theoretic Performance Bounds and Passivity Analysis

Daniel Zelazo, *Member, IEEE*, and Mehran Mesbahi

***Abstract*—This work explores the properties of the edge variant of the graph Laplacian in the context of the *edge agreement* problem. We show that the edge Laplacian, and its corresponding agreement protocol, provides a useful perspective on the well-known *node agreement*, or the consensus algorithm. Specifically, the dynamics induced by the edge Laplacian facilitates a better understanding of the role of certain subgraphs, e.g., cycles and spanning trees, in the original agreement problem. Using the edge Laplacian, we proceed to examine graph-theoretic characterizations of the $\mathcal{H}_2$ and $\mathcal{H}_\infty$ performance for the agreement protocol. These results are subsequently applied in the contexts of optimal sensor placement for consensus-based applications. Finally, the edge Laplacian is employed to provide new insights into the nonlinear extension of linear agreement to agents with passive dynamics.**

***Index Terms*—Agreement protocol, consensus, networked dynamic systems, passivity.**

I. Introduction

COORDINATION and control of multi-agent systems has become an increasingly active area of research in the systems and control community. Applications of such systems are far reaching and include problems such as formation flying, coordinated robotics, sensor fusion, and distributed computation [2]–[6], [15], [20], [21], [27], [19]. An important sub-class of these problems is known as the consensus, or agreement protocol. In consensus problems, agents in a distributed system are able to agree on a common value of interest via a local interaction protocol. There is a wealth of literature on consensus problems, and the reader is referred to the survey papers [23], [28], and the references therein for a more detailed treatment of the subject.

A fundamental theme of consensus problems is the convergence properties of the protocol to a common value based on the structural properties of the underlying graph. In this direction, it is well known that the second smallest eigenvalue of the graph Laplacian matrix, referred to as the *algebraic connectivity* of a graph [10], is a judicious measure for relating the rate of convergence of the protocol to the connectedness of the graph. In fact, the importance of the Laplacian spectra for convergence analysis has been explored in many contexts, including random networks [13], switching topologies [24], and noisy networks [32].

While convergence properties may arguably be the most critical feature of consensus-type problems, a more control theoretic approach to the analysis and synthesis of these systems is also needed. Examples of such studies include a Nyquist-based stability analysis for consensus-based feedback systems [9], a graph-centric notion of controllability in consensus problems [26], and consensus algorithms with guaranteed $\mathcal{H}_\infty$ performance [18]. An important theme in these approaches is the development of a clear connection between certain properties of the underlying connection topology and the system-theoretic properties that these networked systems exhibit.

In this work, we focus on developing a systematic framework for exploring connections between the properties of the network on one hand, and notions from systems and control on the other, in the setting of consensus problems. In this direction, we first develop an edge variant of the graph Laplacian which we term the *edge Laplacian* [33]. The edge Laplacian is a matrix representation of the underlying network that offers a more transparent understanding of how certain graph structures, such as spanning trees and cycles, relate to the algebraic properties of the corresponding matrix. We show that this alternative representation relates to the graph Laplacian matrix via an appropriate similarity transformation. The importance of this matrix representation has been implicitly realized in the literature [1], [8], [22], [31]; one of our contributions in this paper is to explicitly highlight the insights that this matrix representation offers in the analysis and synthesis of multi-agent networks.

The edge Laplacian is the primary construct used to consider the *edge agreement protocol*. In consensus problems, the exchange of information is dependent on the relative information between neighboring agents. This relative information has a natural interpretation in terms of the edges in the interconnection graph, and a corresponding dynamic description of the consensus protocol can be written from the edge perspective. Using the edge agreement model, we are then able to perform a system theoretic analysis of the system's performance using both $\mathcal{H}_2$ and $\mathcal{H}_\infty$ norms. To provide a broader scope for this model, we also introduce exogenous inputs in the form of process and sensor noise. The analysis highlights the role of spanning trees and cycles and also discusses the ramification of the analysis in the context of k-regular graphs. Subsequently, the utility of the proposed framework is demonstrated via examples including $\mathcal{H}_2$ optimal sensor selection for consensus-type systems and nonlinear consensus over networks consisting of passive agents.

The organization of this paper is as follows. In Section I-A, a brief overview of our notation and a few concepts from algebraic

Manuscript received May 18, 2009; revised January 06, 2010; accepted June 28, 2010. Date of publication July 08, 2010; date of current version March 09, 2011. This work was supported by the National Science Foundation Grants ECS-0501606 and CMMI-0856737. Recommended by Associate Editor M. Prandini.

D. Zelazo is with the Institute for Systems Theory and Automatic Control, Universität Stuttgart, Stuttgart 70550, Germany (e-mail: daniel.zelazo@ist.uni-stuttgart.de).

M. Mesbahi is with the Department of Aeronautics and Astronautics, University of Washington, Seattle, WA, 98195-2400, USA (e-mail: mesbahi@uw.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TAC.2010.2056730

graph theory are presented. These constructs are then used to develop and introduce the edge Laplacian in Section II-A. The edge agreement problem is then presented in Section II-C. In Section III, an $\mathcal{H}_2$ and $\mathcal{H}_\infty$ performance analysis of the agreement protocol is presented. Applications of the edge Laplacian for passive networks is then given in Section IV, followed by our concluding remarks in Section V.

A. Preliminaries and Notation

The matrix-theoretic notation used in the paper is as follows: for a matrix A, $\mathcal{R}(A)$ and $\mathcal{N}(A)$ denote, respectively, its range space and null space. Diagonal matrices will be written as $D = \mathbf{diag}\{d_1, \ldots, d_n\}$, with d_i denoting the i-th entry on the diagonal. A matrix and/or a vector that consists of all zero entries will be denoted by $\mathbf{0}$; whereas, "0" will simply denotes the scalar zero. Similarly, the vector $\mathbf{1}$ denote the vector of all ones, and $\mathbf{J} = \mathbf{1}\mathbf{1}^T$. The notation $g(n) = \mathcal{O}(f(n))$ signifies that the function $g(n)$ is bounded from above by some constant multiple of $f(n)$ for large enough values of n. The set of real numbers will be denoted as $\mathbb{R}$, and $\|\cdot\|_p$ denotes the p-norm of its argument (e.g., $p = 2, \infty$), which will be used for vectors, matrices, and system norms. For the set S, $|S|$ denotes its cardinality.

Graphs and the matrices associated with them will be extensively used in this work. The reader is referred to [11] for a detailed treatment of the subject and we present here only a minimal summary of relevant constructs and results.

An undirected (simple) graph $\mathcal{G}$ is specified by a vertex set $\mathcal{V}$ and an edge set $\mathcal{E}$ whose elements characterize the incidence relation between distinct pairs of $\mathcal{V}$. The notation $i \sim j$ is used to denote that node i is connected to node j, or equivalently, $e = (i, j) \in \mathcal{E}$. We make use of the $|\mathcal{V}| \times |\mathcal{E}|$ incidence matrix, $E(\mathcal{G})$, for a graph with an arbitrary orientation, i.e., a graph whose edges have a head (terminal node) and a tail (an initial node). The columns of $E(\mathcal{G})$ are then indexed by the edge set, and the i-th row entry takes the value "1" if it is the initial node of the corresponding edge, "-1" if it is the terminal node, and zero otherwise. From the definition of the incidence matrix it follows that the null space of its transpose, $\mathcal{N}(E(\mathcal{G})^T)$, contains the *agreement subspace,* $\mathbf{span}\{\mathbf{1}\}$. More generally, $\mathbf{span}\{A\}$ for the matrix A will be used to denote the subspace generated by linear combinations of its columns. The rank of the incidence matrix depends only on $|\mathcal{V}|$ and the number of its connected components [11]. The diagonal matrix $\Delta(\mathcal{G})$ of the graph contains the degree of each vertex on its diagonal. The adjacency matrix, $A(\mathcal{G})$, is the symmetric $|\mathcal{V}| \times |\mathcal{V}|$ matrix with zero on the diagonal and one in the ij-th position if node i is adjacent to node j. The (graph) Laplacian of $\mathcal{G}$

$$L(\mathcal{G}) := E(\mathcal{G})E(\mathcal{G})^T = \Delta(\mathcal{G}) - A(\mathcal{G}) \tag{1.1}$$

is a rank deficient positive semi-definite matrix. The eigenvalues are real and will be ordered and denoted as $0 = \lambda_1(\mathcal{G}) \leq \lambda_2(\mathcal{G}) \leq \cdots \leq \lambda_{|\mathcal{V}|}(\mathcal{G})$.

A connected graph $\mathcal{G}$ can be written as the union of two edge-disjoint subgraphs on the same vertex set as $\mathcal{G} = \mathcal{G}_\tau \cup \mathcal{G}_c$, where $\mathcal{G}_\tau$ is a spanning tree subgraph and $\mathcal{G}_c$ contains the remaining edges that necessarily complete the cycles in $\mathcal{G}$. Similarly, the columns of the incidence matrix for the graph $\mathcal{G}$ can always be permuted such that $E(\mathcal{G})$ can be written as

$$E(\mathcal{G}) = [\, E(\mathcal{G}_\tau) \quad E(\mathcal{G}_c)\,]. \tag{1.2}$$

The cycle edges can be constructed from linear combinations of the tree edges via a linear transformation [29], [30], as

$$E(\mathcal{G}_\tau)T_\tau^c = E(\mathcal{G}_c) \tag{1.3}$$

where

$$T_\tau^c = (E(\mathcal{G}_\tau)^T E(\mathcal{G}_\tau))^{-1} E(\mathcal{G}_\tau)^T E(\mathcal{G}_c). \tag{1.4}$$

Using (1.3) we obtain the following alternative representation of the incidence matrix of the graph:

$$E(\mathcal{G}) = E(\mathcal{G}_\tau)\,[\, I \quad T_\tau^c\,] = E(\mathcal{G}_\tau)R(\mathcal{G}) \tag{1.5}$$

the rows of the matrix

$$R(\mathcal{G}) = [\, I \quad T_\tau^c\,] \tag{1.6}$$

are viewed as the basis for the *cut space* of $\mathcal{G}$ [11]. The matrix $[\, -T_\tau^c \quad I\,]^T$, on the other hand, forms a basis for the *flow space*.

The matrix $R(\mathcal{G})$ (1.6), which will play an important role in the present work, has a close connection with a number of structural properties of the underlying network. For example, the number of spanning trees in a graph, $\tau(\mathcal{G})$, can be determined from the cut space basis [11], as

$$\tau(\mathcal{G}) = \det\,[R(\mathcal{G})R(\mathcal{G})^T]. \tag{1.7}$$

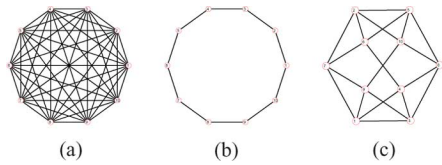

Fig. 1. Example of regular graphs. (a) K_{10} graph (b) C_{10} graph (c) A 4-regular graph.

In order to apply the framework developed in this paper to specific graphs, we will work with the complete graph and its generalization in terms of k-regular graphs, which are defined as follows. The *complete graph* on n nodes, K_n, is the graph where all possible pairs of vertices are adjacent, or equivalently, if the degree of all vertices is $|\mathcal{V}| - 1$. Fig. 1(a) depicts K_{10}, the complete graph on 10 nodes. When every node in a graph with n nodes has the same degree $k \leq n - 1$, it is called a k-regular graph. The k-regular graph on n nodes for $k = 2$ is called the *cycle graph*, C_n. Fig. 1(b) and (c) show, respectively, the cycle graph C_{10} and a 4-regular graph. The line graph of $\mathcal{G}$, denoted as $\mathcal{L}(\mathcal{G})$, is the graph where the edges of $\mathcal{G}$ correspond to the nodes of $\mathcal{L}(\mathcal{G})$, and two edges in $\mathcal{L}(\mathcal{G})$ are adjacent if they share a node in $\mathcal{G}$.

The edges in a graph can be given orientations, which not only facilitate defining the incidence matrix of a graph, but also notions of the cut and the flow spaces of a graph. We define two

edges $e_k, e_l \in \mathcal{E}$ to be positively (negatively) adjacent if they share a node and point in opposite (same) directions relative to this shared node. We denote the positive and negative adjacency of edges as $e_k \sim^+ e_l$ and $e_k \sim^- e_l$, respectively. By convention, an edge is not considered adjacent to itself.

II. The Edge Perspective

In this section, we define a matrix representation of graphs that will be used to highlight the contribution of the edges on the evolution of the consensus protocol. This is then followed by an edge variant of the agreement protocol that we refer to as the edge agreement protocol.

A. Edge Laplacian

We first introduce the edge Laplacian and present some graph theoretic and algebraic interpretations of its structure. In this venue, we will draw parallels between the graph and edge Laplacians whenever possible. The *edge Laplacian* is defined as [33]

$$L_e(\mathcal{G}) := E(\mathcal{G})^T E(\mathcal{G}). \tag{2.8}$$

The structure of the edge Laplacian is closely related to the adjacency matrix of the undirected *line* graph of $\mathcal{G}$, $\mathcal{L}(\mathcal{G})$, as

$$A(\mathcal{L}(\mathcal{G})) = |L_e(\mathcal{G}) - 2I| \tag{2.9}$$

where $|B|$ for a matrix B denotes its entry-wise absolute value. We note that the non-zero eigenvalues of $L_e(\mathcal{G})$ are identical to those of $L(\mathcal{G})$ [14], and each independent cycle in $\mathcal{G}$ corresponds to an eigenvalue at zero in $L_e(\mathcal{G})$. The notion of an independent cycle can be related algebraically to the number of independent columns in $E(\mathcal{G}_c)$. Furthermore, the null space of the edge Laplacian depends on the number of cycles in the graph. Let us elaborate on this last statement via a few definitions and observations.

Definition 2.1: Given an incidence matrix $E(\mathcal{G})$ for a directed graph, a *signed path vector* is a vector $z \in \mathbb{R}^{|\mathcal{E}|}$ corresponding to a path such that the ith element of z takes the value "$+1$" if edge i is traversed positively, "-1" if traversed negatively, and "0" if the edge is not used in the path.

Lemma 2.2: Given a path with distinct initial and terminal nodes described by a signed path vector z in a graph $\mathcal{G}$, the vector $y \in \mathbb{R}^{|\mathcal{V}|}$ is defined as $y = E(\mathcal{G})z$ and the ith element of y takes the value "$+1$" if node i is the initial node of the path, "-1" if it is the terminal node of the path, and "0" otherwise.

Proof: The proof follows directly from the structure of the incidence matrix and Definition 2.1. ∎

Theorem 2.3 [11]: Given a connected graph $\mathcal{G}$ with arbitrary orientation assigned, the null space of $E(\mathcal{G})$ is spanned by all the linearly independent signed path vectors corresponding to the cycles in $E(\mathcal{G})$.

Proof: For any node used in a cycle, the path must enter and exit that node an equal number of times. It then follows from the structure of the incidence matrix that $E(\mathcal{G})z = \mathbf{0}$ when z is the signed path vector for a cycle. ∎

Theorem 2.3 is an example of the intricate relationship between the graphical and algebraic properties of a graph.

Theorem 2.4: Let $L_e(\mathcal{G})$ and $E(\mathcal{G})$ denote, respectively, the edge Laplacian and the incidence matrix of the graph $\mathcal{G}$. Then

$$\mathcal{N}(L_e(\mathcal{G})) = \mathcal{N}(E(\mathcal{G})). \tag{2.10}$$

Proof: Let $x \in \mathcal{N}(E)$; then $L_e(\mathcal{G})x = E(\mathcal{G})^T E(\mathcal{G})x = \mathbf{0}$ and it follows that $\mathcal{N}(E(\mathcal{G})) \subseteq \mathcal{N}(L_e)$. On the other hand when $x \in \mathcal{N}(L_e(\mathcal{G}))$, $E(\mathcal{G})^T E(\mathcal{G})x = \mathbf{0}$ and $x^T E(\mathcal{G})^T E(\mathcal{G})x = \|E(\mathcal{G})x\|_2^2 = 0$. Thus $x \in \mathcal{N}(E(\mathcal{G}))$. ∎

We can alternatively define the edge Laplacian in an analogous way to the identity (1.1). In this venue, let us define the edge adjacency matrix as

$$[A_e(\mathcal{G})]_{kl} := \begin{cases} +1 & e_k \sim^+ e_l \\ -1 & e_k \sim^- e_l \\ 0 & \text{otherwise.} \end{cases}$$

The edge degree matrix, $\Delta_e(\mathcal{G})$, is a diagonal matrix with the number of nodes connected to each edge. As we do not allow self-loops, we have that $\Delta_e(\mathcal{G}) = 2I$. Thus, the edge Laplacian can be equivalently defined as

$$L_e(\mathcal{G}) := 2I - A_e(\mathcal{G}). \tag{2.11}$$

This alternative definition can be used to further deepen the connection between the edge Laplacian of $\mathcal{G}$ and its line graph $\mathcal{L}(\mathcal{G})$. We note that the (i,i) element of $A_e^2(\mathcal{G})$ corresponds to the degree of each node in the line graph of $\mathcal{G}$.

B. Similarity Between the Graph and Edge Laplacians

The connection between the edge and graph Laplacians can be made explicit through the introduction of an appropriate similarity transformation. Furthermore, we find similarity transformations that relate the Laplacians for connected graphs with cycles to graphs on spanning trees. The following theorems assume $\mathcal{G}_\tau \subseteq \mathcal{G}$ is a spanning tree subgraph of $\mathcal{G}$, and $R(\mathcal{G})$ is defined via (1.5).

Theorem 2.5: The graph Laplacian for a connected graph $L(\mathcal{G})$ containing cycles is similar to

$$\begin{bmatrix} L_e(\mathcal{G}_\tau)R(\mathcal{G})R(\mathcal{G})^T & 0 \\ 0 & 0 \end{bmatrix}.$$

Proof: We define the transformation

$$S_v(\mathcal{G}) = \begin{bmatrix} E(\mathcal{G}_\tau)(E(\mathcal{G}_\tau)^T E(\mathcal{G}_\tau))^{-1} & \mathbf{1} \end{bmatrix}$$
$$S_v(\mathcal{G})^{-1} = \begin{bmatrix} E(\mathcal{G}_\tau)^T \\ \left(\frac{1}{|\mathcal{V}|}\right)\mathbf{1}^T \end{bmatrix}.$$

Applying the transformation $S_v(\mathcal{G})^{-1} L(\mathcal{G}) S_v(\mathcal{G})$ leads to the desired result. ∎

Theorem 2.5 provides a transparent way to separate the zero eigenvalue of the Laplacian for a connected graph while preserving algebraic properties of the graph via the edge Laplacian.

Theorem 2.6: The edge Laplacian for a graph with cycles, $L_e(\mathcal{G})$ is similar to the matrix

$$\begin{bmatrix} L_e(\mathcal{G}_\tau)R(\mathcal{G})R(\mathcal{G})^T & 0 \\ 0 & \mathbf{0} \end{bmatrix}$$

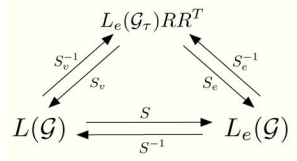

Fig. 2. Transformations between the node and edge Laplacians.

where the block-matrix of zeros is square with dimension equal to the number of independent cycles in the graph.

Proof: Define the transformation matrix

$$S_e(\mathcal{G}) = \begin{bmatrix} R(\mathcal{G})^T & V_e(\mathcal{G}) \end{bmatrix}$$
$$S_e(\mathcal{G})^{-1} = \begin{bmatrix} (R(\mathcal{G})R(\mathcal{G})^T)^{-1}R(\mathcal{G}) \\ V_e(\mathcal{G})^T \end{bmatrix}$$

where $V_e(\mathcal{G})$ is the matrix representation of the orthonormal basis for the null space of $L_e(\mathcal{G})$. As shown by Theorem 2.3, the columns of $V_e(\mathcal{G})$ span the cycle space of the underlying graph. Applying the transformation $S_e(\mathcal{G})^{-1}L_e(\mathcal{G})S_e(\mathcal{G})$ leads to the desired result. ∎

Theorem 2.6 shows that the eigenvalues of $L_e(\mathcal{G}_\tau)R(\mathcal{G})R(\mathcal{G})^T$ correspond to the non-zero eigenvalues of $L_e(\mathcal{G})$. We also note that the block matrix of zeros is square of size equal to the dimension of the kernel of $L_e(\mathcal{G})$. The above results can now be combined to characterize a similarity transformation between the graph and edge Laplacians.

Theorem 2.7: The edge Laplacian for a graph, $L_e(\mathcal{G})$, is similar to the bordered graph Laplacian

$$\begin{bmatrix} L(\mathcal{G}) & 0 \\ 0 & \mathbf{0} \end{bmatrix}$$

where the block-matrix of zeros is square with dimension equal to the number of independent cycles in the graph minus one.

Proof: We define the transformation

$$S(\mathcal{G}) = S_e(\mathcal{G})\bar{S}_v(\mathcal{G})^{-1} \tag{2.12}$$

where

$$\bar{S}_v(\mathcal{G}) = \begin{bmatrix} S_v(\mathcal{G}) & 0 \\ 0 & I \end{bmatrix}$$

and I is the identity matrix of the size of dimension of $\mathcal{N}(L_e(\mathcal{G}))$ minus one. ∎

Theorem 2.7 highlights an important transformation between the graph and edge Laplacians. In both representations, the algebraic structure of the graph is retained while emphasizing the role of spanning trees. Note that when $\mathcal{G} = \mathcal{G}_\tau$ (no cycles), then $R = I$ and we see a direct connection between the graph and edge Laplacians. Furthermore, for a connected graph $\mathcal{G}_\tau$, the edge Laplacian is guaranteed to be invertible as all its eigenvalues are strictly positive. Fig. 2 shows a graphical representation of the relationship between the edge and graph Laplacians.

C. Edge Agreement

In this section we derive an edge variant of the agreement or the consensus protocol using the edge Laplacian introduced in Section II-A. One of the goals of this section is to develop an input-output description of the consensus protocol in order to derive the $\mathcal{H}_2$ and $\mathcal{H}_\infty$ performance of the system which we will delve into in Section III.

The consensus model is built upon a general setup consisting of a group of n identical single integrator units each connected to a fixed number of other units in the ensemble [23]. We generalize the setup by introducing a zero-mean Gaussian process noise, $w_i(t)$, with $\mathbf{E}[w(t)w(t)^T] = \sigma_w^2 I$, to each agent as

$$\dot{x}_i(t) = u_i(t) + w_i(t). \tag{2.13}$$

Labeling these units as 1 through n, the interconnection between the dynamic units can be represented by a graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ with $\mathcal{V} = \{1, \ldots, n\}$ and $\mathcal{E}$ denoting the set of pairwise inter-unit couplings. The interaction or coupling between units' dynamics is realized through the control input $u_i(t)$ in (2.13), assumed to be the sum of the differences between states of an agent and its neighbors. This can be viewed as a decentralized output feedback control, which in our setup is also corrupted by an uncorrelated zero-mean Gaussian measurement noise, $v_{ij}(t)$, with $\mathbf{E}[v(t)v(t)^T] = \sigma_v^2 I$, as

$$u_i(t) = \sum_{i \sim j} (x_j(t) - x_i(t) + v_{ij}(t));$$
$$u(t) = -E(\mathcal{G})^T x(t) + v(t). \tag{2.14}$$

Expressing the dynamic evolution of the resulting system in a compact matrix form with $x(t) = [x_1(t), \ldots, x_n(t)]^T$, one has for the noise-free case

$$\dot{x}(t) = -L(\mathcal{G})x(t) \tag{2.15}$$

where $x(t)$ denotes the collection of node states.

Recall that the agreement set $\mathcal{A} \subseteq \mathbb{R}^n$ is the subspace $\mathbf{span}\{\mathbf{1}\}$. Let us also define $\delta(t)$ as the projection of states $x(t)$ onto the subspace orthogonal to the agreement subspace. This subspace will be denoted by $\mathbf{1}^\perp$; in [24] it is referred to as the *disagreement* subspace. It then follows that $\delta(t) = x(t) - \alpha\mathbf{1}$, where $\alpha = (1/n)\sum_i x_i(0)$.

Proposition 2.8 ([24]): The Laplacian dynamics (2.15) converges to the agreement subspace from an arbitrary initial condition if and only if the underlying graph is connected.

Equation (2.13) along with the relative measurements can be considered as the *open-loop consensus model*. We denote this open-loop system as

$$\Sigma_{ol} : \begin{cases} \dot{x}(t) = u(t) + w(t) \\ y(t) = E(\mathcal{G})^T x(t) + v(t) \end{cases}. \tag{2.16}$$

When the output-feedback control $u(t) = -E(\mathcal{G})y(t)$ is applied, the system leads to a generalized consensus protocol with noise. The noisy consensus model will be referred to as the Σ model specified by

$$\Sigma : \begin{cases} \dot{x}(t) = -L(\mathcal{G})x(t) + \begin{bmatrix} I & -E(\mathcal{G}) \end{bmatrix} \begin{bmatrix} w(t) \\ v(t) \end{bmatrix} \\ z(t) = E(\mathcal{G})^T x(t). \end{cases} \tag{2.17}$$

In (2.17), the variable $z(t)$ is introduced as a monitored performance signal. The open-loop system is shown in Fig. 3 with the consensus output-feedback law.

This set-up has a natural "edge interpretation" that we now examine. In this direction, we introduce the coordinate transformation $S_v x_e(t) = x(t)$, where S_v is defined in (2.12). Applying

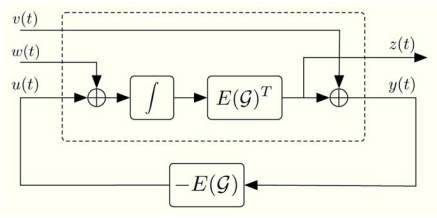

Fig. 3. Open-loop consensus system with output feedback.

this transformation to the consensus system with noise (2.17) yields

$$\Sigma_e : \begin{cases} \dot{x}_e(t) = \begin{bmatrix} -L_e(\mathcal{G}_\tau)R(\mathcal{G})R(\mathcal{G})^T & 0 \\ 0 & 0 \end{bmatrix} x_e(t) + \\ \qquad \begin{bmatrix} E(\mathcal{G}_\tau)^T & -L_e(\mathcal{G}_\tau)R(\mathcal{G}) \\ \frac{1}{n}\mathbf{1}^T & 0 \end{bmatrix} \begin{bmatrix} w(t) \\ v(t) \end{bmatrix} \\ z(t) = \begin{bmatrix} R(\mathcal{G})^T & 0 \end{bmatrix} x_e(t). \end{cases}$$

The benefit of such a transformation is in view of the preservation of the algebraic structure of the underlying connection topology through the edge Laplacian. Furthermore, we note that the new state $x_e(t)$ can be partitioned as $x_e^T(t) = [\, x_\tau^T(t) \quad x_{\mathbf{1}}^T(t) \,]^T$, where $x_\tau(t)$ represents the relative state information across the edges of a spanning tree of $\mathcal{G}$, and $x_{\mathbf{1}}(t)$ is the mode in the $\mathbf{1}$ subspace; the $x_{\mathbf{1}}(t)$ mode can be interpreted as the "inertial state" for the entire formation. In fact, we note that this transformation separates the system into its controllable and observable parts; that is, the $x_{\mathbf{1}}(t)$ mode is an unobservable mode of the system.

We can now consider a minimal realization of the system containing only the states $x_\tau(t)$ for analysis. We refer to this as the Σ_τ system specified by

$$\Sigma_\tau : \begin{cases} \dot{x}_\tau(t) = -L_e(\mathcal{G}_\tau)R(\mathcal{G})R(\mathcal{G})^T x_\tau(t) + \\ \qquad \sigma_w E(\mathcal{G}_\tau)^T \hat{w}(t) - \sigma_v L_e(\mathcal{G}_\tau)R(\mathcal{G})\hat{v}(t) \\ z(t) = R(\mathcal{G})^T x_\tau(t). \end{cases} \tag{2.18}$$

The signals $\hat{w}(t)$ and $\hat{v}(t)$ are the normalized process and measurement noise signals. The performance variable, $z(t)$, contains information on the tree states in addition to the cycle states. Here we recall that the cycle states are linear combination of the tree states and we note that $z(t)$ actually contains redundant information. This is highlighted by recognizing that the tree states converging to the origin forces the cycle states to do the same. Consequently, we will consider the system with cycles as well as a system containing only the tree states at the output, which we denote as $\hat{\Sigma}_\tau$

$$\hat{\Sigma}_\tau : \begin{cases} \dot{x}_\tau(t) = -L_e(\mathcal{G}_\tau)R(\mathcal{G})R(\mathcal{G})^T x_\tau(t) + \\ \qquad \sigma_w E(\mathcal{G}_\tau)^T \hat{w}(t) - \sigma_v L_e(\mathcal{G}_\tau)R(\mathcal{G})\hat{v}(t) \\ z(t) = x_\tau(t). \end{cases} \tag{2.19}$$

This distinction will subsequently be employed to quantify the effect of cycles on the system performance. In the noise-free case, (2.18) reduces to the edge variant of the autonomous system,

$$\Sigma_\tau : \begin{cases} \dot{x}_\tau(t) = -L_e(\mathcal{G}_\tau)R(\mathcal{G})R(\mathcal{G})^T x_\tau(t) \\ z(t) = R(\mathcal{G})^T x_\tau(t) \end{cases} \tag{2.20}$$

both systems (2.18) and (2.20) are referred to as the *edge agreement protocol*.

The first simple, yet important observation, relates to the meaning of agreement in the context of the edge states, leading to an edge interpretation of Proposition 2.8.

Proposition 2.9: The edge agreement problem (2.20) converges to the origin for arbitrary graphs.

Proof: If the graph $\mathcal{G}$ is connected, then we note that the agreement state is equivalent to having $x_\tau(t) = \mathbf{0}$, as $x_e(t) = S_v^{-1} x(t)$ for $x(t) \in \mathcal{A}$. In the edge setting, the agreement set $\mathcal{A}_e$, maps to the *origin*. The projection of the edge states onto this set, denoted as $\delta_e(t)$, is consequently the norm of the edge states; it also satisfies $\|\delta_e(t)\|_2 = \|x_\tau(t)\|_2 \leq \|E(\mathcal{G})\|_2 \|\delta(t)\|_2$ with respect to the distance to the agreement subspace. For a disconnected graph $\mathcal{G}$ with c connected components, we can conclude using the results of Section II-B that each component of the edge agreement system will converge to the origin. ■

From Proposition 2.8, the node dynamics over a connected graph converges to the agreement subspace, which implies that the corresponding edge dynamics converges to the origin. An important consequence of this result is the edge agreement will not always correspond to the node agreement; having all the relative states converge to the origin will not guarantee that each node state has the same value. This merely emphasizes the need to work with connected graphs. Analogous to the node agreement, in the edge agreement setting, the evolution of an edge state depends on its current state and the states of its adjacent edges, i.e., those that share a node with it.

Much of the literature related to consensus problems focuses on the convergence rate of the system; a property dictated by the second smallest eigenvalue of the graph Laplacian. The input-output description of the consensus problem developed in this section allows for a more general notion of performance for these systems. We explore such ramifications next.

III. Graph-Theoretic Performance Bounds

In this section, we delve into the characterization of system performances for the agreement protocol, measured in terms of the $\mathcal{H}_2$ and $\mathcal{H}_\infty$ system norms, using the framework developed in Section II. In this section, we will assume that the underlying interconnection graph is connected; moreover, we will omit the dependency of the matrix R on $\mathcal{G}$ when it is implicit and unambiguous. Furthermore, we introduce the shorthand notations L_e^τ and E_τ in place of $L_e(\mathcal{G}_\tau)$ and $E(\mathcal{G}_\tau)$.

A. $\mathcal{H}_2$ Performance

We first recall that the $\mathcal{H}_2$ performance of the system characterizes how a (Gaussian) exogenous noise propagates throughout the system and effects the energy of the monitored output. In the context of the agreement protocol, therefore, *the $\mathcal{H}_2$ system norm can be employed to reason about how noise on the edges of the network result in the asymptotic deviation of each node's state from the consensus state.* However, the limiting factor for an $\mathcal{H}_2$ analysis of the standard consensus model whose system matrix is the graph Laplacian, is that for any connected graph, the system Σ has an *unbounded* $\mathcal{H}_2$ norm due to the presence of the zero eigenvalue. In this section, we proceed to perform this analysis using the edge agreement protocol.

The $\mathcal{H}_2$ norm of Σ_τ and $\hat{\Sigma}_\tau$ can be calculated as [7]

$$||\Sigma_\tau||_2^2 = \mathbf{trace}[R^T X^* R], \text{ and } ||\hat{\Sigma}_\tau||_2^2 = \mathbf{trace}[X^*] \quad (3.21)$$

where X^* is the positive-definite solution to the Lyapunov equation

$$-L_e^\tau R R^T X - X R R^T L_e^\tau + \sigma_w^2 L_e^\tau + \sigma_v^2 L_e^\tau R R^T L_e^\tau = 0. \quad (3.22)$$

The structure of (3.22) suggests that the solution will be dependent on certain properties of the graph. In fact, the solution can be found by inspection as

$$X^* = \frac{1}{2}\left(\sigma_w^2 (R R^T)^{-1} + \sigma_v^2 L_e^\tau\right) \quad (3.23)$$

and we arrive at the following result.

Theorem 3.1: The $\mathcal{H}_2$ norm of the Σ_τ (2.18) system is

$$||\Sigma_\tau||_2^2 = \frac{\sigma_w^2}{2}(n-1) + \sigma_v^2 |\mathcal{E}|. \quad (3.24)$$

On the other hand, the $\mathcal{H}_2$ norm of the $\hat{\Sigma}_\tau$ system (2.19) is

$$||\hat{\Sigma}_\tau||_2^2 = \frac{\sigma_w^2}{2}\mathbf{trace}[(R R^T)^{-1}] + \sigma_v^2 (n-1). \quad (3.25)$$

Proof: The proof follows from (3.23) and noting that $\mathbf{trace}[L_e^\tau] = 2(n-1)$, or twice the number of edges in a spanning tree. ∎

We observe that $||\Sigma_\tau||_2^2$ is a linear function of the number of edges in the graph. This has a clear practical relevance, as it indicates that the addition of each edge corresponds to an amplification of the noise in the consensus-type network. Let us consider the implications of the graph-theoretic characterization of the $\mathcal{H}_2$ norm for two classes of graphs.

1) Spanning Trees: The first case resulting in a simplification of (3.24) arises when $\mathcal{G}$ is a spanning tree. In this case $R = I$ and (3.25) simplifies to

$$||\hat{\Sigma}_\tau||_2^2 = (n-1)\left(\frac{\sigma_w^2}{2} + \sigma_v^2\right). \quad (3.26)$$

A direct consequence of this result is that *all* spanning trees result in the same $\mathcal{H}_2$ system performance. That is, the choice of spanning tree (e.g., a path or a star) does not affect this performance metric. This is in contrast to results related to the rate of convergence which would favor a tree with a larger $\lambda_2(\mathcal{G})$. As expected, in this scenario $||\Sigma_\tau||_2^2 = ||\hat{\Sigma}_\tau||_2^2$.

2) k-Regular Graphs: Regular graphs also lead to a simplification of (3.25). In general, any connected k-regular graph will contain cycles resulting in a non-trivial expression for the matrix product RR^T. The $\mathcal{H}_2$ norm is therefore intimately related to the cut space of the graph.

Denote the eigenvalues of RR^T by μ_i and note that

$$\mathbf{trace}[(R R^T)^{-1}] = \sum_{i=1}^{n-1} \frac{1}{\mu_i} = \frac{1}{\tau(\mathcal{G})} \sum_{i=1}^{n-1} \prod_{j \neq i}^{n-1} \mu_j$$

where $\tau(\mathcal{G})$ is the number of spanning trees in $\mathcal{G}$. The quantity $\prod_{j\neq i}^{n-1} \mu_j$ is recognized as a first minor of the matrix RR^T.

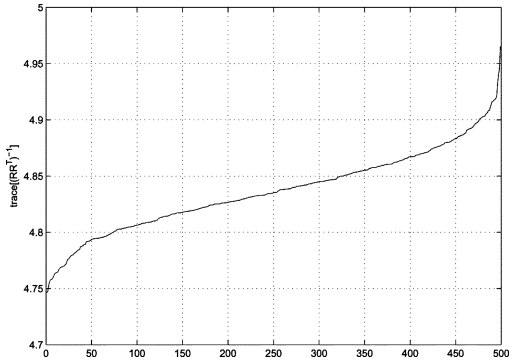

Fig. 4. $\mathbf{trace}[R(\mathcal{G})R(\mathcal{G})^T)^{-1}]$ for random 5-regular graphs.

Corollary 3.2: The cycle graph C_n has n spanning trees and hence

$$\mathbf{trace}\left[(R(C_n)R(C_n)^T)^{-1}\right] = \frac{(n-1)^2}{n}. \quad (3.27)$$

Thereby, the $\mathcal{H}_2$ norm of the $\hat{\Sigma}_\tau$ system when the underlying graph is the cycle graph C_n is given as

$$||\hat{\Sigma}_\tau||_2^2 = (n-1)\left(\frac{\sigma_w^2(n-1)}{n} + \sigma_v^2\right). \quad (3.28)$$

Proof: Without loss of generality, we consider a directed path graph on n nodes, with initial node v_1 and terminal node v_n as the spanning tree subgraph $\mathcal{G}_\tau$. Index the edges as $e_i = (v_i, v_{i+1})$. The cycle graph is formed by adding the edge $e_n = (v_n, v_1)$. For this graph, we have $T_\tau^c = \mathbf{1}_{n-1}$ and $R(C_n)R(C_n)^T = I + \mathbf{J}$. It follows that $\det\left[R(C_n)R(C_n)^T\right] = n$ and all its first minors have value $n-1$. Combined with (3.24) yields the desired result. ∎

Corollary 3.3: The complete graph K_n has n^{n-2} spanning trees, and therefore

$$\mathbf{trace}\left[\left(R(K_n)R(K_n)^T\right)^{-1}\right] = \frac{2(n-1)}{n}. \quad (3.29)$$

Thereby, the $\mathcal{H}_2$ norm of the $\hat{\Sigma}_\tau$ system when the underlying graph is the complete graph K_n is given as

$$||\hat{\Sigma}_\tau||_2^2 = (n-1)\left(\frac{\sigma_w^2}{n} + \sigma_v^2\right). \quad (3.30)$$

Proof: Without loss of generality, we consider a star graph with center at node v_1 and all edges are of the form $e_k = (v_1, v_{k+1})$. Then the cycles in the graph are created by adding the edges $e = (v_i, v_j)$, $i, j \neq 1$ and $R(K_n)R(K_n)^T = nI - \mathbf{J}$. It then follows that:

$$\det\left[R(K_n)R(K_n)^T\right] = n^{n-2}$$

and all the first minors have value $2n^{n-3}$. Combined with (3.24) yields the desired result. ∎

Fig. 4 depicts the sorted values of $\mathbf{trace}\left[\left(RR^T\right)^{-1}\right]$ for 500 randomly generated regular graphs of degree five. As this figure shows, although the degree of each node remains constant, the actual cycle structure of each graph instance varies, effecting the resulting $\mathcal{H}_2$ norm of the corresponding consensus-type input-output system.

Using the above analysis, we now proceed to characterize how the cycle structure of the graph effects the $\mathcal{H}_2$ performance for the corresponding consensus-type system. In fact, examining the ratio

$$\frac{||\Sigma_\tau(\mathcal{G})||_2^2}{||\Sigma_\tau(\mathcal{G}_\tau)||_2^2}$$

provides an indication of how the cycles increase the $\mathcal{H}_2$ norm; recall that $\mathcal{G}$ is in general a graph containing cycles and $\mathcal{G}_\tau \subseteq \mathcal{G}$ is the spanning tree subgraph.

For example, consider the cycle graph C_n and assume unit covariance for both the process and measurement noises. Then, as the number of nodes increase, the ratio of the two $\mathcal{H}_2$ norms behaves as

$$\lim_{n\to\infty} \frac{||\Sigma_\tau(C_n)||_2^2}{||\Sigma_\tau(P_n)||_2^2} = \lim_{n\to\infty} \frac{4n-2}{3n} = \frac{4}{3} \tag{3.31}$$

indicating that for large cycles, the $\mathcal{H}_2$ performance is a constant multiple of the $\mathcal{H}_2$ performance for the path graph P_n.

In the meantime, for the complete graph K_n we have

$$\frac{||\Sigma_\tau(K_n)||_2^2}{||\Sigma_\tau(\mathcal{G}_\tau)||_2^2} = \frac{n+1}{3} = \mathcal{O}(n) \tag{3.32}$$

in this case, we see that the norm is amplified linearly as a function of the number of vertices in the graph. It is worth mentioning here that typical performance measures for consensus problems, such as $\lambda_2(\mathcal{G})$, would favor the complete graph over the cycle graph. However, in terms of the $\mathcal{H}_2$ performance, we see that there is a penalty to be paid for faster convergence offered by the complete graph due to its cycle structure.

Alternatively, insight is also gained by considering the ratio

$$\frac{||\Sigma_\tau(\mathcal{G})||_2^2}{||\hat{\Sigma}_\tau(\mathcal{G})||_2^2}$$

which highlights the effects of including cycles in the performance variable $z(t)$.

For the cycle graph we have

$$\lim_{n\to\infty} \frac{||\Sigma_\tau(C_n)||_2^2}{||\hat{\Sigma}_\tau(C_n)||_2^2} = \lim_{n\to\infty} \frac{n(3n-1)}{2(n-1)(2n-1)} = \frac{3}{4} \tag{3.33}$$

suggesting that the effect of including the cycle for performance does not vary significantly with the size of the graph.

For the complete graph, on the other hand, one has

$$\frac{||\Sigma_\tau(K_n)||_2^2}{||\hat{\Sigma}_\tau(K_n)||_2^2} = \frac{n}{2} = \mathcal{O}(n) \tag{3.34}$$

suggesting that the inclusion of cycles results in $\mathcal{H}_2$ performance that increases linearly as a function of vertices in the graph.

B. Sensor Placement With $\mathcal{H}_2$ Performance

Encouraged by the graph-theoretic characterization of the $\mathcal{H}_2$ performance for consensus-type systems, in this section, we proceed to consider the problem of sensor selection and placement for consensus-type systems. Consider, for example, a scenario where there are two types of sensors available for the relative measurements in the open-loop consensus problem. One sensor is high-fidelity and high cost, with associated noise covariance of $\overline{\sigma}_v^2$. The other sensor is a less expensive lower fidelity sensor with covariance of $\underline{\sigma}_v^2 > \overline{\sigma}_v^2$. When synthesizing the topology for the consensus problem, the designer must consider the tradeoff between the sensor costs and the overall system performance.

In this direction, consider the system in (2.18) in the form

$$\Sigma_\tau : \begin{cases} \dot{x}(t) = -L_e^\tau R R^T x_\tau(t) + \sigma_w E_\tau^T \hat{w}(t) - L_e^\tau R \Gamma \hat{v}(t) \\ z(t) = R^T x_\tau(t) \end{cases} \tag{3.35}$$

where $\hat{w}(t)$ and $\hat{v}(t)$ are the normalized noise signals, and the matrix Γ is a diagonal matrix with elements σ_i corresponding to the variance of the sensor on edge i. We note that the most general version of this problem considers a finite set of p sensors each with an associated variance

$$P = \{\sigma_1^2, \sigma_2^2, \ldots, \sigma_p^2\} \tag{3.36}$$

where for each element $\sigma_i^2 \in P$ there is an associated cost $c(\sigma_i^2)$. The cost function has the property that $c(\sigma_i^2) > c(\sigma_j^2)$ if $\sigma_i^2 < \sigma_j^2$. Using (3.21)–(3.22), in order to find the optimal placement of these sensors, one can consider the mixed-integer program [17]

$$\begin{aligned} \mathcal{P}_1 : \quad & \min_{X,W} \quad \lambda \mathbf{trace}[R^T X R] + \sum_{i=1}^{|\mathcal{E}|} c(w_i) \\ \text{s.t.} \quad & W = \mathbf{diag}\{w_1, \ldots, w_{|\mathcal{E}|}\},\ w_i \in P, \sum_i w_i \le \mu, \\ & -L_e^\tau R R^T X - X R R^T L_e^\tau \\ & +\sigma_w^2 L_e^\tau + L_e^\tau R W R^T L_e^\tau = 0 \end{aligned}$$

where λ represents a weighting on the $\mathcal{H}_2$ performance of the solution, and μ represents the maximum aggregated noise covariance. Note that in general $|\mathcal{E}| \min_i \sigma_i^2 \le \mu \le |\mathcal{E}| \max_i \sigma_i^2$.

The problem $\mathcal{P}_1$ is combinatorial in nature, as a discrete decision must be made on the placement and type of sensor in the network. Although $\mathcal{P}_1$ can certainly be solved by using mixed-integer programming solvers [17], certain relaxations can be made to convexify the resulting problem. Most notably, one approach involves relaxing the discrete nature of the set P (3.36) into a box-type constraint as

$$\hat{P} = \left[\underline{\sigma}^2, \overline{\sigma}^2\right]. \tag{3.37}$$

The cost function can now be written as a continuous map $c : \hat{P} \mapsto \mathbb{R}$ which is convex and a strictly decreasing function. The simplest version of such a function would be the linear map, $c(\sigma_i^2) = -\beta \sigma_i^2$, for some $\beta > 0$. This relaxation leads to the following modified program

$$\begin{aligned} \mathcal{P}_2 : \quad & \min_{X,W} \quad \lambda \mathbf{trace}[R^T X R] - \beta \mathbf{trace}[W] \\ \text{s.t.} \quad & W = \mathbf{diag}\{w_1, \ldots, w_{|\mathcal{E}|}\},\ w_i \in \hat{P}, \sum_i w_i \le \mu, \\ & -L_e^\tau R R^T X - X R R^T L_e^\tau \\ & +\sigma_w^2 L_e^\tau + L_e^\tau R W R^T L_e^\tau = 0. \end{aligned}$$

As an example of the applicability of $\mathcal{P}_2$, we considered the sensor selection for the graph in Fig. 5. A random graph on 10 nodes with an edge probability of 0.15 was generated. The resulting graph is connected and contains two independent cycles, resulting in a more general problem instance. The sensor constraints were $\hat{P} = [\,0.001 \quad 0.1\,]$ and $\mu^2 = 0.501$. Finally, the

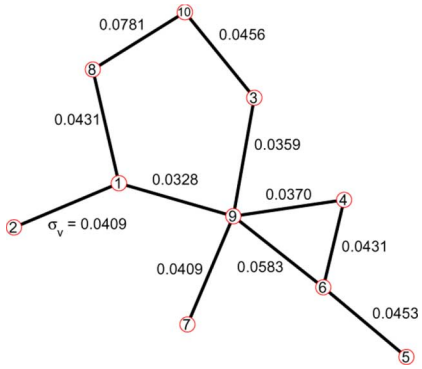

Fig. 5. Graph on 10 nodes with optimal sensor selection; σ_v denotes the sensor variance.

cost function weights were chosen as $\beta = 5$ and $\lambda = 1$. Solving $\mathcal{P}_2$ then resulted in a non-trivial selection of sensors for each edge. The sensor covariance for each edge is labeled in Fig. 5; we observe that the highest fidelity sensors tend to be concentrated around the node of highest degree. Also, the edge with the lowest fidelity sensor is placed in "low traffic" areas.

C. $\mathcal{H}_\infty$ Performance

We first recall that the $\mathcal{H}_\infty$ norm for a dynamic system captures how a measurable signal with finite energy, i.e., a signal in $\mathcal{L}_2$, is amplified at the monitored output of the system. Moreover, this norm has implications for robustness, disturbance rejection, and uncertainty management for dynamic systems. Specifically, the $\mathcal{H}_\infty$ norm of a linear system with transfer-function representation $H(s)$ can be characterized as

$$\|H(s)\|_\infty = \sup_\omega \{\overline{\sigma}[H(j\omega)]\} \tag{3.38}$$

where $\overline{\sigma}[A]$ denotes the largest singular value of the matrix A. In the context of the agreement protocol, therefore, *the $\mathcal{H}_\infty$ system norm can be used to capture how disturbances and finite energy exogenous signals, including reference signals, result in the asymptotic deviation of each node state from consensus.* In this section, in view of (2.18), we proceed to examine the $\mathcal{H}_\infty$-norm for the agreement protocol using an edge perspective.

To begin this analysis, we first write the transfer-function representation of (2.18) as

$$\Sigma_\tau(s) = R^T \left(sI + L_e^\tau R R^T\right)^{-1} \left[\sigma_w E_\tau^T \quad -\sigma_v L_e^\tau R\right]. \tag{3.39}$$

The transfer-function representation for $\hat{\Sigma}_\tau(s)$ is similarly defined from its state-space representation. Before we begin our analysis of the transfer-function matrix (3.39), let us provide a useful result on the ordering properties of the eigenvalues of congruent Hermitian matrices.

Theorem 3.4 ([14]): Let A be a Hermitian matrix and S a nonsingular square matrix. Let the eigenvalues of A and SS^* be arranged in an increasing order. Then, for each $k = 1, \ldots, n$, there exists a positive real number θ_k such that $\lambda_1(SS^*) \leq \theta_k \leq \lambda_n(SS^*)$ and

$$\lambda_k(SAS^*) = \theta_k \lambda_k(A). \tag{3.40}$$

Recall now that the state matrix $-L_e^\tau R R^T$ in (2.18) arises from a similarity transformation with the graph Laplacian, as shown in (2.12). This allows us to infer that the eigenvalues of $L_e^\tau R R^T$ are all positive and real, and the state matrix is diagonalizable. Therefore, we can diagonalize the system using a modal decomposition with transformation matrix S to obtain

$$\begin{aligned} \dot{\hat{x}}_\tau(t) &= -\Lambda(\mathcal{G})\hat{x}_\tau(t) + S^{-1}\left[\sigma_w E_\tau^T \quad -\sigma_v L_e^\tau R\right] \begin{bmatrix} \hat{w}(t) \\ \hat{v}(t) \end{bmatrix} \\ z(t) &= R^T S \hat{x}_\tau(t) \end{aligned} \tag{3.41}$$

where $\Lambda(\mathcal{G}) = \mathbf{diag}\{\lambda_2(\mathcal{G}), \ldots, \lambda_n(\mathcal{G})\}$.

Consider first a variation of (3.41) where the output equation is simplified to $z(t) = \hat{x}_\tau(t)$. The modified system has a transfer matrix representation

$$H(s) = (sI + \Lambda(\mathcal{G}))^{-1} B \tag{3.42}$$

where for notational simplicity we have defined $B := S^{-1}\left[\sigma_w E_\tau^T \quad -\sigma_v L_e^\tau R\right]$, noting that $\Sigma_\tau(s) = R^T S H(s)$.

Proposition 3.5: For the system matrix $H(s)$ in (3.42), one has

$$\|H(s)\|_\infty = \overline{\sigma}(H(0)). \tag{3.43}$$

Proof: From (3.38), we must find the singular values of $H(j\omega)$. This is facilitated by examining the eigenvalues of $H(j\omega)H^*(j\omega)$. Defining $Q = \Lambda(\mathcal{G})^{-1} B B^T \Lambda(\mathcal{G})^{-1}$ and $V(\omega) = (j\omega\Lambda(\mathcal{G})^{-1} + I)^{-1}$ we have $H(j\omega)H^*(j\omega) = V(\omega)QV^*(\omega)$.

We note that the last identity describes $H(j\omega)H^*(j\omega)$ as a congruence transformation of the matrix Q, which is in the form required to use Theorem 3.4. The matrix $V(\omega)V^*(\omega)$ has the form

$$V(\omega)V^*(\omega) = \mathbf{diag}\left\{\frac{\lambda_2(\mathcal{G})^2}{\omega^2 + \lambda_2(\mathcal{G})^2}, \ldots, \frac{\lambda_n(\mathcal{G})^2}{\omega^2 + \lambda_n(\mathcal{G})^2}\right\}. \tag{3.44}$$

Denote the eigenvalues of $V(\omega)V^*(\omega)$ as $\mu_i(\omega) = \lambda_{i+1}(\mathcal{G})^2/(\omega^2 + \lambda_{i+1}(\mathcal{G})^2)$ to highlight their dependency on the frequency ω. It is now verified that for any fixed frequency ω, we have $\mu_1(\omega) \leq \mu_2(\omega) \leq \cdots \leq \mu_n(\omega)$. Furthermore, for any $\omega > 0$, we have $\mu_i(\omega) < 1$ for $i = 1, 2, \ldots, n-1$. We thereby invoke Theorem 3.4 to conclude that $\theta_k(\omega) < 1$ for $k = 1, 2, \ldots, n-1$ for all $\omega > 0$. At $\omega = 0$, $H(j\omega)H^*(j\omega) = Q$ and hence the singular values of $H(j\omega)$ are a strictly decreasing function of ω. Therefore, the maximum singular value must occur at $\omega = 0$ which corresponds to $\overline{\sigma}(H(0))$, concluding the proof. ■

It remains to show that introducing the output equation $z(t) = R^T S \hat{x}_\tau(t)$ does not change the frequency at which the supremum in (3.38) occurs.

Proposition 3.6: The $\mathcal{H}_\infty$-norm for the system (3.41) corresponds to the maximum singular value of its transfer matrix at $\omega = 0$.

Proof: It suffices to show that $\|R^T S H(s)\|_\infty = \overline{\sigma}(R^T S H(0))$, where $H(s)$ is defined in (3.42). The system $H(s)$ has a singular value decomposition $H(s) = U\Sigma(s)V^*$, with $U \in \mathbb{R}^{n-1 \times n-1}$ and $V \in \mathbb{R}^{n+|\mathcal{E}| \times n+|\mathcal{E}|}$. Consider a pure sinusoidal input $W(j\omega)$ expressed in terms of the basis vectors in V as

$$W(j\omega) = \sum_{i=1}^{n+|\mathcal{E}|} \alpha_i(j\omega) v_i$$

where v_i is the i-th column of V. We can express the output of $H(j\omega)$ to the sinusoidal input as

$$Y(j\omega) = H(j\omega) \sum_{i=1}^{n+|\mathcal{E}|} \alpha_i(j\omega) v_i = \sum_{i=1}^{n+|\mathcal{E}|} \alpha_i(j\omega)\sigma_i(j\omega)u_i.$$

Similarly, the matrix $R^T S$ has a singular value decomposition PMG^*, with $P \in \mathbb{R}^{n-1\times n-1}$ and $G \in \mathbb{R}^{|\mathcal{E}|\times|\mathcal{E}|}$. As $R^T S$ is connected in series with $H(s)$, we can express the output of the overall system as

$$\begin{aligned} Z(j\omega) &= R^T S H(j\omega) W(j\omega) \\ &= \sum_{i=1}^{n+|\mathcal{E}|} \alpha_i(j\omega)\sigma_i(j\omega) \left(\sum_{j=1}^{|\mathcal{E}|} \beta_j \mu_j p_j \right) \\ &= \sum_{i=1}^{n+|\mathcal{E}|} \alpha_i(j\omega)\sigma_i(j\omega)\xi_i \end{aligned} \tag{3.45}$$

where we have expressed each signal as a linear combination of the appropriate basis vectors, and μ_i is the i-th singular value of $R^T S$.

For $\|W(j\omega)\|_2 = 1$, the $\mathcal{H}_\infty$ norm of $R^T S H(s)$ is equivalently characterized by finding the frequency that maximizes $\|Z(j\omega)\|_2$. Using (3.45), we express the output norm at a given frequency as

$$\|Z(j\omega)\|_2^2 = \sum_{i=1}^{n+|\mathcal{E}|} \left(|\alpha_i(j\omega)\sigma_i(j\omega)|\right)^2 \|\xi_i\|_2^2$$

where we used the property that $\xi_i^T \xi_j = 0$ for $i \neq j$.

As we are restricting the input to be on the unit ball, we have that $\sum |\alpha_i(j\omega)|^2 = 1$. From Proposition 3.5 we have that $|\sigma_i(j\omega)| < |\sigma_i(0)|$ for all $\omega > 0$. Therefore, it is straightforward to verify that the coefficients $|\alpha_i(j\omega)\sigma_i(j\omega)|$ are maximized at $\omega = 0$. ∎

Using the above observations, we now state a general result on the $\mathcal{H}_\infty$-norm of the edge agreement system.

Theorem 3.7: The $\mathcal{H}_\infty$ norms for Σ_τ (2.18) and $\hat{\Sigma}_\tau$ (2.19), are, respectively

$$\|\Sigma_\tau\|_\infty^2 = \sigma_w^2 \left(\overline{\sigma} \left[R^T (R R^T L_e^\tau R R^T)^{-1} R \right] \right) + \sigma_v^2 \tag{3.46}$$

and

$$\|\hat{\Sigma}_\tau\|_\infty^2 = \overline{\sigma} \left[\sigma_w^2 (R R^T L_e^\tau R R^T)^{-1} + \sigma_v^2 (R R^T)^{-1} \right]. \tag{3.47}$$

Proof: From Propositions 3.5 and 3.6, we can evaluate (3.39) at $s = 0$ and calculate the singular values of the corresponding matrix as

$$\Sigma_\tau(s)|_{s=0} = R^T (L_e^\tau R R^T)^{-1} \begin{bmatrix} \sigma_w E_\tau^T & \sigma_v L_e^\tau R \end{bmatrix}.$$

In general, $\Sigma_\tau(s)$ is not square, so we determine the singular values by finding the eigenvalues of $\Sigma_\tau(s)\Sigma_\tau(s)^*$. In this direction, we observe that

$$\begin{aligned} \Sigma_\tau(s)\Sigma_\tau(s)^*|_{s=0} = \sigma_w^2 R^T \left(R R^T L_e^\tau R R^T \right)^{-1} R \\ + \sigma_v^2 R^T (R R^T)^{-1} R. \end{aligned} \tag{3.48}$$

We note that the second term in (3.48) is a projection matrix. Moreover, the matrix $R^T (R R^T)^{-1} R$ has exactly $n-1$ eigenvalues at one, and the remaining eigenvalues at zero (with multiplicity equal to the number of independent cycles). As R has full row rank, and L_e^τ and $R R^T$ are invertible matrices, we have that both terms in (3.48) have the same null space. Therefore, the eigenvalues of $\Sigma_\tau(s)\Sigma_\tau(s)^*|_{s=0}$ can be determined from the first term in (3.48) which yields the desired result. For the $\hat{\Sigma}_\tau$ system, an analogous proof can used by replacing the observation matrix R^T with identity. ∎

As in Section III-A, we provide examples on how for certain classes of graphs the expression (3.46) can be simplified and interpreted.

1) Spanning Trees: When the underlying graph is a spanning tree, we have that $R = I$, and (3.47) reduces to

$$\|\hat{\Sigma}_\tau\|_\infty^2 = \sigma_w^2 \left(\overline{\sigma} \left[(L_e^\tau)^{-1} \right] \right) + \sigma_v^2. \tag{3.49}$$

In the context of $\mathcal{H}_\infty$, we see that the choice of spanning tree is important, as opposed to the corresponding scenario for the $\mathcal{H}_2$ norm. In [25] and [12] it was shown that the path graph has the smallest largest eigenvalue of the graph Laplacian, and the star graph has the greatest largest eigenvalue. As $\|\hat{\Sigma}_\tau\|_\infty$ is determined by the inverse of the edge Laplacian, we conclude that the star graph corresponds to the tree topology with minimum $\mathcal{H}_\infty$ norm, and the path graph with largest norm. As in the $\mathcal{H}_2$-norm case, we have $\|\Sigma_\tau\|_\infty = \|\hat{\Sigma}_\tau\|_\infty$.

2) k-Regular Graphs: As shown in Section III-A, regular graphs admit certain algebraic simplifications that prove useful for system norm calculations for the corresponding agreement protocol. To maintain a parallel analysis with the $\mathcal{H}_2$ problem, we examine the cycle graph and complete graph as special cases here. First, we note that the only spanning tree subgraph for the cycle graph, C_n, is the path graph, P_n. We therefore state some results on the path graph.

Proposition 3.8: The edge Laplacian for the path graph P_n is the tridiagonal matrix

$$L_e(P_n) = \begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{bmatrix}. \tag{3.50}$$

Proof: The structure of (3.50) follows from the edge adjacency definition of the edge Laplacian (2.11). This structure also assumes an ordering of nodes and edges such that node i is always connected to node $i+1$ by edge e_i. ∎

Proposition 3.9: The inverse of the edge Laplacian for the path graph P_n is determined by observing that

$$\left[(L_e(P_n))^{-1} \right]_{ij} = \frac{\min(i,j)(n - \max(i,j))}{n}. \tag{3.51}$$

Proposition 3.10: For the cycle graph C_n with $\mathcal{G}_\tau = P_n$

$$R(C_n)^T (R(C_n) R(C_n)^T)^{-1} = \begin{bmatrix} I - \frac{1}{n}\mathbf{J} \\ -\frac{1}{n}\mathbf{1}^T \end{bmatrix}. \tag{3.52}$$

Proof: For the cycle graph, it was shown in Section III-A that $R(C_n)R(C_n)^T = I + \mathbf{J}$. It then follows that $(R(C_n)R(C_n)^T)^{-1} = I - (1/n)\mathbf{J}$. ∎

Proposition 3.11: The matrix

$$R(C_n)^T(R(C_n)R(C_n)^T L_e(P_n)R(C_n)R(C_n)^T)^{-1}R(C_n)$$

is similar to

$$\begin{bmatrix} (L_e(P_n)R(C_n)R(C_n)^T)^{-1} & 0 \\ 0 & 0 \end{bmatrix}. \tag{3.53}$$

Proof: Defining the transformation matrix

$$M = \begin{bmatrix} L_e(P_n)R(C_n) \\ \mathbf{1}^T \end{bmatrix}$$
$$M^{-1} = [\, R(C_n)^T(R(C_n)R(C_n)^T)^{-1}(L_e(P_n))^{-1} \quad \tfrac{1}{n}\mathbf{1}\,]$$

yields the desired result. ∎

We note that the eigenvalues of the matrix (3.53) is $(1/\lambda_i(C_n))$ for $i = 2, \ldots, n$, i.e., these eigenvalues are the inverse of the non-zero eigenvalues of the graph Laplacian for the cycle graph.

Corollary 3.12: The $\mathcal{H}_\infty$-norm of the Σ_τ system (2.18) when the underlying graph is the cycle graph C_n is given as

$$\|\Sigma_\tau\|_\infty^2 = \frac{\sigma_w^2}{\lambda_2(C_n)} + \sigma_v^2 \tag{3.54}$$

where $\lambda_2(C_n)$ denotes the second smallest eigenvalue of the cycle graph Laplacian.

Proof: The proof follows from Theorem 3.7 and Proposition 3.11. ∎

Proposition 3.13: The edge Laplacian for the star graph S_n is specified as

$$[L_e(S_n)]_{ij} = \begin{cases} 2, & i = j \\ 1, & i \neq j \end{cases}. \tag{3.55}$$

Proof: The structure of (3.55) follows from the edge adjacency definition of the edge Laplacian (2.11) by noting that every edge is adjacent to each other in the star graph. This structure also assumes each edge is positively adjacent to each other edge. ∎

Proposition 3.14: The inverse of the edge Laplacian for the path graph S_n is

$$(L_e(S_n))^{-1} = I - \frac{1}{n}\mathbf{J}. \tag{3.56}$$

Corollary 3.15: The $\mathcal{H}_\infty$ norm of the Σ_τ system when the underlying graph is the complete graph K_n is given as

$$\|\Sigma_\tau\|_\infty^2 = \frac{\sigma_w^2}{n} + \sigma_v^2. \tag{3.57}$$

The $\mathcal{H}_\infty$ norm of the $\hat{\Sigma}_\tau$ system when the underlying graph is the complete graph K_n is given as

$$\|\hat{\Sigma}_\tau\|_\infty^2 = \frac{\sigma_w^2}{n} + \sigma_v^2. \tag{3.58}$$

Proof: The proof follows from propositions 3.13, 3.14, and the fact that $RR^T = nI - \mathbf{J}$ for the complete graph. ∎

We conclude this section by noting that for the system $\hat{\Sigma}_\tau$ (2.19), one has $\|\hat{\Sigma}_\tau\|_\infty^2 = (n-1)\|\hat{\Sigma}_\tau\|_2^2$.

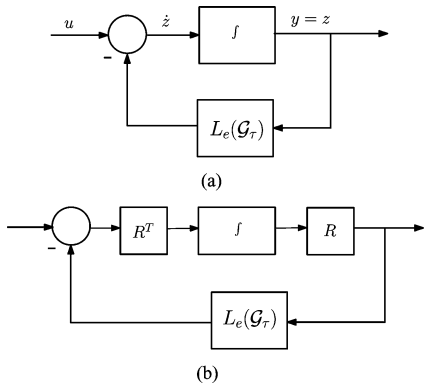

Fig. 6. Viewing edge agreement over a spanning tree as a strictly output passive system.

IV. Edge Laplacian and Nonlinear Agreement

In this section, we consider the application of the edge Laplacian Section II-A for non-linear variations on the basic agreement protocol. This includes viewing the protocol in the context of Lyapunov theory as opposed to the machinery of LaSalle's invariance principle [16]. We then proceed to examine the nonlinear extensions of the agreement problem via the passivity framework [1]. Our contribution in this section is to streamline the analysis of the nonlinear consensus-type problems using the edge Laplacian.

To begin our analysis, consider the noise-free version of the edge agreement problem (2.18). Of course, an underlying assumption of this edge variant of the consensus problem is the linearity of the interaction rule. A natural generalization of this model is to introduce non-linear passive elements in the general setup. As we move from a linear to non-linear model, we explore how passivity theory, in conjunction with the edge Laplacian, can be used to analyze this extension. First, we recall that passivity pertains to nonlinear system of the form

$$\dot{z}(t) = f(z(t), u(t)), \quad y(t) = z(t) \tag{4.59}$$

where f is locally Lipschitz and $f(0,0) = 0$; then (4.59) is passive if there exists a continuously differentiable positive semidefinite function V, referred to as the storage function, such that

$$u(t)^T y(t) \geq \dot{V}(t) \tag{4.60}$$

for all t. If $\dot{V}$ in (4.60) can be replaced by $\dot{V} + \psi(z)$ for some positive definite function ψ, then we call the system strictly passive; in our case, since the output of the system is its state, (4.59) could also be referred to as output strictly passive.

Theorem 4.1 ([16]): Suppose that (4.59) is output strictly passive with a radially unbounded storage function. Then the origin is globally asymptotically stable.

To demonstrate the utility of this passivity theorem in the context of agreement protocol, consider the interconnection of Fig. 6(a), with an integrator in the forward path and the edge Laplacian of a spanning tree, in the feedback path, depicting the edge agreement (2.18). Note that $z(t)$ in this case denotes the

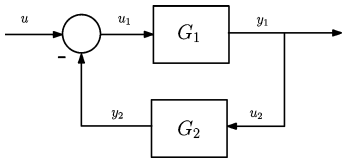

Fig. 7. The feedback configuration for theorem 4.3.

vector of edge states $x_\tau(t)$. Then, with respect to the quadratic storage function $V(x) = (1/2)x^T(t)x(t)$, one has

$$u^T(t)y(t) = u^T(t)z(t) = \dot{V} + z^T(t)L_e(\mathcal{G}_\tau)z(t)$$

implying that the system is strictly (output) passive with a storage function that is radially unbounded. This observation, in turn, makes the convergence analysis for the edge agreement over a spanning tree fall in the range of applicability of Theorem 4.1. Hence, $z(t) \to 0$ as $t \to \infty$ and convergence to the agreement subspace of the "node" states follows, providing an alternative proof for Proposition 2.9.

The connection between the agreement protocol and Theorem 4.1 can be used to extend the basic setup of the agreement protocol in various directions, one of which is the following.

Corollary 4.2: Suppose that for a network of interconnected agents the edge states evolve according to $\dot{x}_\tau(t) = -f(\mathcal{G}, x_\tau(t))$ where $f : \mathcal{G} \times \mathbb{R}^m \to \mathbb{R}^m$ for which $x_\tau^T(t)f(\mathcal{G}, x_\tau(t)) > 0$ for all $x_\tau(t) \neq 0$ when $\mathcal{G}$ is connected. Then the corresponding node states converge to the agreement subspace.

The above corollary suggests that many passivity-type results from nonlinear systems theory can now be applied to the agreement protocol in its edge context.

Corollary 4.3: Consider the feedback connection shown in Fig. 7, where the time-invariant passive system $G_1 : \dot{z}(t) = f(z(t), u_1(t)), y_1(t) = z(t)$ has a storage function V and the time invariant memory-less function G_2 is such that $u_2(t)^T y_2(t) \geq u_2(t)^T \phi(u_2(t))$ for some positive definite function ϕ. Then the origin of the closed loop system (with $u(t) = 0$) is asymptotically stable.

To illustrate the ramification of Corollary 4.3, suppose that following the integrator block in Fig. 6, there exists a nonlinear operator ψ such that for some positive-definite functional $V(z)$, one has $\psi(z) = \nabla V(z)$. Then

$$\dot{V}(t) = \nabla V^T \dot{z}(t) = \psi(z)^T \dot{z}(t) \tag{4.61}$$

implying that the forward path of the feedback configuration shown in Fig. 8(a) is passive with a storage function V and the function $\phi(v)$ in Corollary 4.3 can be chosen as $\lambda_2(\mathcal{G})v$. Hence, the asymptotic stability of the origin with respect to the edge states $x_\tau(t)$ can be implied by invoking Corollary 4.3. The more general case of this result for a connected network is also immediate in light of the identity (1.5) implying that $L_e(\mathcal{G}) = R(\mathcal{G})^T L_e(\mathcal{G}_\tau) R(\mathcal{G})$, where $\mathcal{G}_\tau$ is a spanning tree of the graph $\mathcal{G}$. This relationship suggests the loop transformation depicted in Fig. 8, keeping in mind that passivity of the forward path does not change under post- and pre-multiplication by matrices R^T and R, and the linearity of the integrator operator allows an operator reordering; Corollary 4.3 can now be invoked under this more general setting.

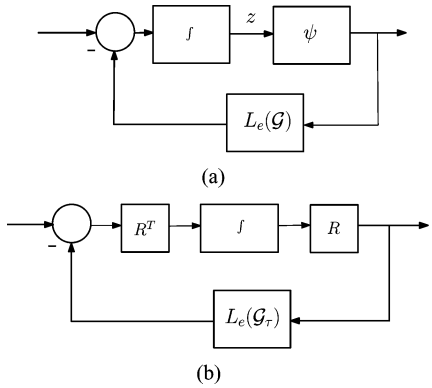

Fig. 8. Loop transformation between feedback connection with edge Laplacian over arbitrary connected graphs shown in (a) to one over spanning trees, shown in (b).

An example that demonstrates the utility of the above observation for multi-agent systems pertains to the Kuramoto model of n-coupled oscillators interacting over the network $\mathcal{G}$ as

$$\dot{\theta}_i(t) = k \sum_{j \sim i} \sin(\theta_j(t) - \theta_i(t)), \quad i = 1, 2, \ldots, n. \tag{4.62}$$

In (4.62) the constant k denotes the coupling strength between the oscillators, which for the purpose of this section is assumed to be positive. The nonlinear interaction rule (4.62) can compactly be represented as

$$\dot{\theta}(t) = -kE(\mathcal{G}) \sin\left(E(\mathcal{G})^T \theta(t)\right)$$

where $\theta(t) = [\theta_1(t), \theta_2(t), \ldots, \theta_n(t)]^T$ and

$$\sin(\theta(t)) = [\sin(\theta_1(t)), \sin(\theta_2(t)), \ldots, \sin(\theta_n(t))]^T.$$

After multiplying $E(\mathcal{G})^T$ on both sides of the above equation, we obtain

$$\dot{\theta}_e(t) = -kL_e(\mathcal{G}) \sin(\theta_e(t)) \tag{4.63}$$

which monitors the relative phases between the oscillators with $\theta_e(t) = E(\mathcal{G})^T \dot{\theta}(t)$ for all t. In order to mold the stability analysis of the Kuramoto model (4.63) in the context of passivity theory, we write

$$\dot{\theta}_e(t) = -kR(\mathcal{G})^T L_e(\mathcal{G}_\tau) R(\mathcal{G}) \sin(\theta_e(t))) \tag{4.64}$$

where $L_e(\mathcal{G}_\tau)$ is the edge Laplacian of a spanning tree of $\mathcal{G}$, and hence a positive definite matrix, or when viewed as a dynamic system, a strictly passive element. Now we let $V(\theta_e(t)) = \mathbf{1}^T(\mathbf{1} - \cos(\theta_e(t)))$ be a candidate storage function for the Kuramoto model (4.64). In this case, $V(\theta_e(t)) > 0$ for all nonzero $\theta_e \pmod{2\pi}$, $V(0) = 0$, and $\psi(\theta_e(t)) = \sin(\theta_e(t))$, in reference to the identity (4.61) and Fig. 8(b). Using the passivity machinery, combined with the edge Laplacian formalism, we thus conclude that for the Kuramoto model over a connected graph, the synchronization state is asymptotically stable.[1] Finally, we note that the transformation used to arrive at (4.63) is not invertible. While an invertible transformation may be used, as in Section II-C, the transformation employed here facilitates the use of the presented passivity approach.

[1]By synchronization we refer to the case when $\theta_1 = \theta_2 = \cdots = \theta_n \bmod 2\pi$.

V. Conclusion

In this paper, we defined and explored the interpretation of an edge variant of the graph Laplacian in the context of the edge agreement problem and a number of its extensions. The results presented in this work not only point to close connections between the well-studied node agreement problem and its edge version, but argues that the edge formulation of the agreement problem, lead to insights in the $\mathcal{H}_2$, $\mathcal{H}_\infty$ and the nonlinear agreement problems. We also showed how the edge Laplacian highlights the role of other features of the network, complementary to its spectral properties, in the performance of the agreement protocol and its extensions.

Acknowledgment

The authors wish to thank the reviewers for their suggestions and A. Rahmani for his many insights and fruitful discussions on the edge Laplacian and the edge agreement protocol.

References

[1] M. Arcak, "Passivity as a design tool for group coordination," *IEEE Trans. Autom. Control*, vol. 52, no. 8, pp. 1380–1390, Aug. 2007.
[2] B. Bamieh, F. Paganini, and M. Dahleh, "Distributed control of spatially-invariant systems," *IEEE Trans. Autom. Control*, vol. 47, no. 7, pp. 1091–1107, Jul. 2002.
[3] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation*. Englewood Cliffs, NJ: Prentice-Hall, 1989.
[4] B-D Chen and S. Lall, "Dissipation inequalities for distributed systems on graphs," in *Proc. IEEE Conf. Decision Control*, Dec. 2003, pp. 3084–3090.
[5] J. Cortes and F. Bullo, "Coordination and geometric optimization via distributed dynamical systems," *SIAM J. Control Optim.*, vol. 45, no. 5, pp. 1543–1574, 2005.
[6] R. D'Andrea and G. E. Dullerud, "Distributed control design for spatially interconnected systems," *IEEE Trans. Autom. Control*, vol. 48, no. 9, pp. 1478–1495, Sep. 2003.
[7] G. E. Dullerud and F. Paganini, *A Course in Robust Control Theory*. New York: Springer, 2005.
[8] D. V. Dimarogonas and K. H. Johansson, "On the stability of distance-based formation control," in *Proc. IEEE Conf. Decision Control*, Dec. 2008, pp. 1200–1205.
[9] 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.
[10] M. Fiedler, "Algebraic connectivity of graphs," *Czechoslovak Math. J.*, vol. 23, no. 98, pp. 298–305, 1973.
[11] C. Godsil and G. Royle, *Algebraic Graph Theory*. New York: Springer, 2001.
[12] I. Gutman, "The star is the tree with greatest greatest Laplacian eigenvalue," *Kragujevac J. Math.*, vol. 24, pp. 61–65, 2002.
[13] Y. Hatano and M. Mesbahi, "Agreement over random networks," *IEEE Trans. Autom. Control*, vol. 50, no. 11, pp. 1867–1872, Nov. 2005.
[14] R. A. Horn and C. Johnson, *Matrix Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 1985.
[15] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Trans. Autom. Control*, vol. 48, no. 6, pp. 988–1001, Jun. 2003.
[16] H. K. Khalil, *Nonlinear Systems*. Englewood Cliffs, NJ: Prentice Hall, 2001.
[17] B. Korte and J. Vygen, *Combinatorial Optimization: Theory and Algorithms*. New York: Springer, 2007.
[18] P. Lin, Y. Jia, and L. Li, "Distributed robust $\mathcal{H}_\infty$ consensus control in directed networks of agents with time-delaty," *Syst. Control Lett.*, vol. 57, pp. 643–653, 2007.
[19] M. Mesbahi and M. Egerstedt, *Graph Theoretic Methods in Multiagent Networks*. Princeton, NJ: Princeton Univ. Press, 2010.
[20] M. Mesbahi and F. Y. Hadaegh, "Formation flying control of multiple spacecraft via graphs, matrix inequalities, and switching," *AIAA J. Guid., Control, Dyn.*, vol. 24, no. 2, pp. 369–377, 2001.
[21] L. Moreau, "Stability of multiagent systems with time-dependent communication links," *IEEE Trans. Autom. Control*, vol. 50, no. 2, pp. 169–182, Feb. 2005.
[22] A. Muhammad and M. Egerstedt, "Control using higher order Laplacians in network topologies," in *Proc. Math. Theory Netw. Syst.*, Jul. 2006, pp. 1024–1038.
[23] 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.
[24] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Trans. Autom. Control*, vol. 49, no. 9, pp. 1520–1533, Sep. 2004.
[25] M. Petrovic and I. Gutman, "The path is the tree with smallest greatest Laplacian eigenvalue," *Kragujevac J. Math.*, vol. 24, pp. 67–70, 2002.
[26] A. Rahmani, M. Ji, M. Mesbahi, and M. Egerstedt, "Controllability of multi-agent systems from a graph theoretic perspective," *SIAM J. Control Optim.*, vol. 48, no. 1, pp. 162–186, 2009.
[27] W. Ren and R. W. Beard, *Distributed Consensus in Multi-Vehicle Cooperative Control*. London, U.K.: Springer-Verlag, 2008.
[28] W. Ren, R. W. Beard, and E. M. Atkins, "A survey of consensus problems in multi-agent coordination," in *Proc. Amer. Control Conf.*, Jun. 2005, vol. 3, pp. 1859–1864.
[29] J. Sandhu, M. Mesbahi, and T. Tsukamaki, "Cuts and flows in relative sensing and control of spatially distributed systems," in *Proc. Amer. Control Conf.*, Jun. 2005, pp. 73–78.
[30] J. Sandhu, M. Mesbahi, and T. Tsukamaki, "Relative sensing networks: Observability, estimation, and the control structure," in *Proc. IEEE Conf. Decision Control*, Dec. 2005, pp. 6400–6405.
[31] A. Tahbaz-Salehi and A. Jadbabaie, "Distributed coverage verifcation in sensor networks without location information," in *Proc. 47th IEEE Conf. Decision Control*, Dec. 2008, pp. 4170–4176.
[32] L. Xiao, S. Boyd, and S. Lall, "A scheme for robust distributed sensor fusion based on average consensus," in *Proc. 4h Int. Symp. Inform. Processing Sensor Networks*, Apr. 2005, pp. 64–70.
[33] D. Zelazo, A. Rahmani, and M. Mesbahi, "Agreement via the edge Laplacian," in *Proc. IEEE Conf. Decision Control*, Dec. 2007, pp. 2309–2314.

Daniel Zelazo (M'10) was born in Milwaukee, WI, in 1977. He received the B.Sc. and M.Eng. degrees in electrical engineering from the Massachusetts Institute of Technology, Cambridge, in 1999 and 2001, respectively, and the Ph.D. degree in aeronautics and astronautics from the University of Washington, Seattle, in 2009.

In between his Master's and Ph.D. degrees he worked at Texas Instruments, Japan, on audio compression algorithms. He is now a Research Associate at the Universität Stuttgart, Germany, with research interests in networked dynamic systems, optimization, and graph theory.

Mehran Mesbahi received the Ph.D. degree from the University of Southern California, Los Angeles, in 1996.

He was a member of the Guidance, Navigation, and Analysis Group, Jet Propulsion Labs (JPL), from 1996 to 2000 and an Assistant Professor of aerospace engineering and mechanics at University of Minnesota, Minneapolis, from 2000 to 2002. He is currently a Professor of aeronautics and astronautics at the University of Washington, Seattle. His research interests are distributed and networked aerospace systems, systems and control theory, and engineering applications of optimization and combinatorics.

Dr. Mesbahi received the NSF CAREER Award in 2001, the NASA Space Act Award in 2004, the UW Distinguished Teaching Award in 2005, and the UW College of Engineering Innovator Award for Teaching in 2008.