Consensus and Eigenvalue Estimation in Dynamic Consensus Networks with Strictly Positive Real Systems

Fadel Lashhab, Member, IEEE
Department of Electrical Engineering and Computer Science
College of Engineering and Architecture
Howard University, Washington DC, USA.
fadel.lashhab@howard.edu

***Abstract*—The use of consensus networks in networking has received great attention due to its wide array of applications in fields such as robotics, transportation, sensor networking, communication networking, biology, and physics. The focus of this paper is to study a generalization of consensus problems whereby the weights of network edges are no longer static gains, but instead are dynamic systems, leading to the notion of dynamic consensus networks.. Specifically, we consider networks whose nodes are transfer functions (typically integrators) and whose edges are strictly positive real transfer functions representing dynamical systems that couple the nodes. We transform each concept of static graph theory into dynamic terms, out of which a generalized dynamic graph theory naturally emerges. We present a framework for dynamic graphs and dynamic consensus networks. This framework introduces the idea of dynamic degree, adjacency, incident, and Laplacian matrices in a way that naturally extends these concepts from the static case. We show that strictly positive realness of the edges is a sufficient condition for dynamic networks to be stable (i.e., to reach consensus). To study the spectral properties of dynamic networks, we introduce the Dynamic Grounded Laplacian matrix, which is used to estimate lower and upper bounds for the real parts of the smallest and largest non-zero eigenvalues of the dynamic Laplacian matrix. These bounds can be used to obtain stability conditions using the Nyquist graphical stability test for undirected dynamic networks controlled using distributed controllers. Numerical simulations are provided to verify the effectiveness of the results.**

***Keywords*—Dynamic graphs, dynamic Laplacian, Dynamic Consensus, Grounded Laplacian, strictly-positive real systems, spectrum of a dynamic Laplacian.**

I. Introduction

In recent years, consensus in networks has received considerable attention due to a wide array of applications in fields such as robotics, cooperative unmanned air vehicles, transportation, sensor networks, communication networks, air traffic control, biology, and physics. The focus of this paper is to study a generalization of consensus problems in which the weights of network edges are no longer static gains, but instead are dynamic systems, leading to the notion of *dynamic consensus networks*.

To understand the basic idea of *consensus*, consider the graph or network shown in Fig. 1. This graph consists of three nodes or agents with node variables denoted as v_i, v_j, and v_k and edges e_{ij} connecting the i^{th} and j^{th} nodes. Such graphs can be described as a set of nodes (or vertices) $\mathcal{N} = \{n_i\}$ connected by a set of edges $\mathcal{E} = \{(n_i, n_j) : \ n_i, n_j \in \mathcal{N}\}$. Each edge $\mathcal{E}$ is associated with a flow $f_{ij}(t)$, which in this figure is the same as $f_{ji}(t)$, implied by the bi-directional arrows.

In Fig. 1, we assume the node does some form of processing of its inputs and that edges carry a flow of information or energy from one node to the other. Assuming an external input u_i to each node, then a typical form of the processing in each node is, for example,

$$\dot{v_i}(t) = u_i(t) - \sum_{j \in \mathcal{N}_i} f_{ij}(t) \tag{1}$$

where $\mathcal{N}_i$ is the set of indices for nodes that communicate with node i. This is called the neighborhood of node i and the nodes in this case are considered to be integrators. Note that a popular form for $f_{ij}(t)$ is

$$f_{ij}(t) = \lambda_{ij}(v_i(t) - v_j(t)), \tag{2}$$

where λ_{ij} is a static gain or interconnection weight between node i and node j. With $u_i(t) = 0$, (1) results in the so-called *consensus protocol*

$$\dot{v_i}(t) = -\sum_{j \in \mathcal{N}_i} \lambda_{ij}(v_i(t) - v_j(t)). \tag{3}$$

For the traditional consensus networks, the network topology is static, meaning that there are no dynamics in the interconnections between the nodes ($\lambda_{ij} = constant \geq 0$) and the nodes are assumed to be integrators [24]. Thus, static

Copyright © AJSE Publishers

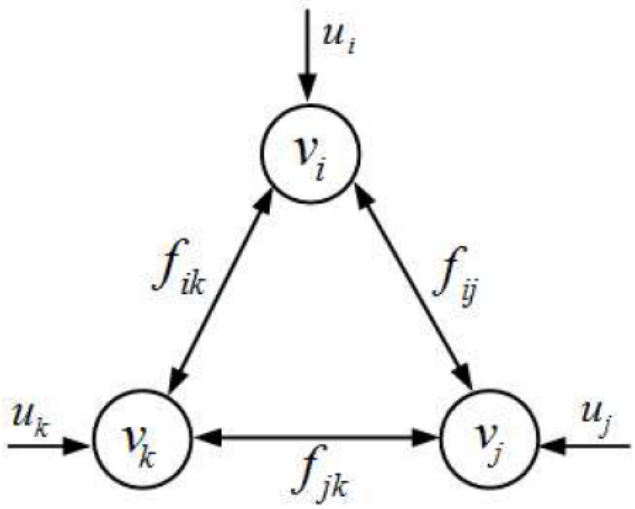


Fig. 1. Graph or network.

consensus problems can be written in the time domain for each node $i = 1, 2, ..., n$ as

$$\dot{x}_i = \sum_{j \in \mathcal{N}_i} \lambda_{ij}(x_j(t) - x_i(t)), \tag{4}$$

where $\mathcal{N}_i$ is called the neighborhood of node i. The continuous-time linear consensus protocol (4) can be written in matrix form as:

$$\dot{v}(t) = -Lv(t), \tag{5}$$

where $v(t) = [v_1(t), v_2(t), ..., v_n(t)]^T$ and $L = [l_{ij}]$, is the Laplacian matrix, defined by

$$l_{ij} = \begin{cases} \sum_{j \in \mathcal{N}_i} \lambda_{ij} & i = j \\ -\lambda_{ij} & i \neq j \text{ and } (i,j) \in \mathcal{E} \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

For the multi-agent consensus problem, the state $v_i \in \mathbb{R}^1$ represents the belief of agent i about a so-called global consensus variable v. These beliefs are updated using communications with their neighbors according to the consensus protocol (4). A key result is that the solution of $\dot{v}(t) = -Lv(t)$ gives $v_i \to v^*$, where v^* is a constant, if the static graph is connected [24]. This specific fact has been the basis of much of the literature related to consensus problems. The focus of this paper is to study a generalization of consensus problems whereby the weights of the network edges are no longer modeled as static gains, but instead are represented as dynamic systems coupling the nodes.

By dynamic systems, we mean that the relationships between the system variables are described by linear ordinary differential equations. We call such networks *dynamic consensus networks* because under some conditions, all the node variables converge to a common value called a consensus value.

Consensus problems using Fig. 1 and its variants have attracted much research interest in recent years, because many applications admit a model in the form of (1), including formation control [8], [17], [20], distributed sensor fusion in sensor networks [9], [25], [34], flocking [4], [31], [32], rendezvous problems in space [1], [19], and synchronization in power networks [6], [21], [26], among others. The origin of formal studies of consensus problems had its roots in the 1960s (see [5] and references therein). A good survey of consensus problems is found in [23], [29], and the references therein.

Recently, several groups have begun to consider consensus over networks with dynamic edges. Such networks arise in applications in several ways. In [33], the authors consider consensus of networks with dynamic channels. In these types of consensus problems, it is assumed that there exists a channel (filter) from node j to node i, making the signal received by i from j to be $\widehat{v}_j(t) = h_{ij}(t) * v_j(t)$, where $h_{ij}(t)$ is the impulse response of a linear, time-invariant (LTI) filter of the edge e_{ij} and "$*$" denotes the convolution operation. By taking the Laplace transform of $\widehat{v}_j(t)$, we can write $\widehat{V}_j(s) = H_{ij}(s)V_j(s)$, where $H_{ij}(s)$ represents the transfer function of the LTI filter of the edge e_{ij}. Based on this, the dynamics of node i for $i = 1, 2, ..., n$ can be written in the time domain as:

$$\dot{v}_i(t) = \sum_{j \in \mathcal{N}_i} \lambda_{ij}[\widehat{v}_j(t) - v_i(t)]. \tag{7}$$

By taking the Laplace transform, we have

$$sV_i(s) - v_i(0) = \sum_{j \in \mathcal{N}_i} \lambda_{ij}[H_{ij}(s)V_j(s) - V_i(s)], \tag{8}$$

which leads us to

$$V(s) = (sI_n + L(s))^{-1}V(0), \tag{9}$$

where $V(s) = [V_1(s), V_2(s), ..., V_n(s)]^T$, $V(0) = [v_1(0), v_2(0), ..., v_n(0)]^T$, and $L(s) = [l_{ij}(s)]$ is the Laplacian matrix defined by:

$$L_{ij}(s) = \begin{cases} \sum_{j \in \mathcal{N}_i} \lambda_{ij} & i = j \\ -\lambda_{ij}H_{ij}(s) & i \neq j \text{ and } (i,j) \in \mathcal{E} \\ 0 & \text{otherwise} \end{cases}. \tag{10}$$

Consensus protocols such as (8) suggest further generalization: Filtering of the difference $(v_i - v_j)$: This type of consensus protocol extends the protocol (8) by filtering the difference between the node variables instead of filtering the difference between the received and transmitted signals. Thus, the dynamic consensus protocol can be written in the time domain for the node i, $i = 1, 2, ..., N$, as

$$\dot{v}_i(t) = \sum_{j \in \mathcal{N}_i} h_{ij}(t) * [v_j(t) - v_i(t)], \tag{11}$$

By taking the Laplace transform of (11), we get

$$sV_i(s) - v_i(0) = \sum_{j \in \mathcal{N}_i} H_{ij}(s)[V_j(s) - V_i(s)], \tag{12}$$

$$V(s) = (sI_n + L(s))^{-1}V(0), \tag{13}$$

Copyright © AJSE Publishers

where $L(s) = [l_{ij}(s)]$ is defined by:

$$l_{ij}(s) = \begin{cases} \sum_{j \in \mathcal{N}_i} H_{ij}(s) & i = j \\ -H_{ij}(s) & i \neq j \text{ and } (i,j) \in \mathcal{E} \\ 0 & \text{otherwise} \end{cases}, \quad (14)$$

$L(s)$ here is different from the Laplacian matrix defined in (10) for the dynamic networks introduced by [33]. We call (14) a dynamic Laplacian matrix. To our knowledge, the model is unique to our research, though note that in a recent paper [30], the filter of the difference $[v_i - v_j]$ also appears.

The consensus protocol given in (4) and its variants have been studied extensively in the literature and have been applied in a number of areas, notably to formation motion control when the agents are mobile. We have been motivated to consider this extension to transfer functions weights from several applications. In particular, the thermal model of heat flows in buildings can be shown to admit a generalized dynamic consensus network over a directed graph [22]. In another example, modeling the load frequency control (LFC) network of an electrical power grid can be viewed as a single-integrator consensus network with dynamic interconnection coefficients $\lambda_{ij} = G_i(s)a_{ij}$ [16]. To further motivate the concept from an application, in the next section we show how an electrical circuit whose nodes are grounded capacitors can be modeled as an undirected dynamic consensus network.

This paper offers several distinct contributions. First, we show that the nodes variables in dynamic consensus networks whose nodes are integrators and whose edges are strictly positive real transfer functions will converge to a common consensus value. Second, we develop methods for estimating the bounds of the eigenvalues of a dynamic Laplacian matrix. This is accomplished by introducing the idea of a grounded dynamic Laplacian sub-matrix as a means of reducing the order of complexity of computation that would be necessary in the case of the full dynamic Laplacian matrix. Third, we use the estimated bounds of the eigenvalues of a dynamic Laplacian to analyze the consensus and stability of a dynamic consensus network under distributed feedback control.

The paper is organized as follows: Section II presents a motivating physical example of a dynamic consensus network. In Section III, we present a framework for dynamic graphs and dynamic consensus networks. This framework introduces the idea of dynamic degree, adjacency, incident, and Laplacian matrices in a way that naturally extends these concepts from the static case. From this, we can easily define equivalent concepts of dynamic interconnection matrices and dynamic consensus networks. Section IV we investigate the consensus conditions of dynamic consensus networks with integrator nodes and strictly positive real (SPR) edges. For such graphs, we can introduce what we call the dynamic Laplacian and dynamic Grounded Laplacian of the dynamic graphs. In Section V, lower and upper bounds for the largest and smallest non-zero eigenvalues of the dynamic Laplacian matrix are derived. Finally, using these bounds, we characterize in Section VI the stability of a closed loop system consisting of an undirected dynamic consensus network controlled by distributed controllers, and a numerical example is provided to illustrate the results.

II. Dynamic Consensus Network

A generalization of static consensus networks can be motivated when the consensus network consists of an interconnection of physical systems (e.g., an electrical network that requires a common operating condition such as voltage or frequency) or in cases when the communication between agents in a consensus network includes dynamic behavior (e.g., delays). This paper studies a generalization of the static consensus problem whereby the weights are no longer positive gains, but instead represent strictly-positive real dynamic systems. The use of positive real (PR) transfer functions is motivated by the study of dynamic networks that represent linear electrical circuits composed of resistors, inductors and capacitors. A strictly positive real transfer (SPR) function corresponds to the driving input impedance in dissipative networks that contain resistors, inductors and capacitors [28].

Definition 2.1: [28] A $n \times n$ proper rational transfer function matrix $G(s)$ is called positive real if any of the following equivalent condition hold:

1) Poles of all elements of $G(s)$ are in $\text{Re}[s] \leq 0$.
2) For all real ω for which $j\omega$ is not a pole of any element of $G(s)$, the matrix $G(j\omega) + G^T(-j\omega)$ is positive semidefinite (PSD).
3) Any pure imaginary pole $j\omega$ of any element of $G(s)$ is a simple pole and the residue matrix $\lim_{s \to j\omega}(s - j\omega)G(s)$ is positive semidefinite Hermitian.

When $n = 1$, the second condition of the definition reduces to $[\text{Re}G(j\omega)] \geq 0$, $\forall \omega \in \mathbb{R}$. In addition, a real rational transfer function $h(s)$ is said to be strictly positive real (SPR) if $h(s-\epsilon)$ is positive real for some $\epsilon > 0$.

Consider a dynamic network whose node equations can be written in the time domain for the node i, $i = 1, 2, ..., n$, as

$$\dot{v}_i(t) = -\sum_{j \in \mathcal{N}_i} y_{ij}(t) * [v_i(t) - v_j(t)], \quad (15)$$

where v_i is a consensus variable of the node i, $y_{ij}(t)$ is the impulse response of the edge e_{ij}, and "$*$" denotes the convolution operation. Taking the Laplace transform of (15) results in

$$V_i(s) = -\frac{1}{s}\sum_{j \in \mathcal{N}_i} Y_{ij}(s)[V_i(s) - V_j(s)] + \frac{1}{s}v_i(0), \quad (16)$$

where $Y_{ij}(s)$ is the Laplace transform of $y_{ij}(t)$. We refer to (15) or (16)as a *dynamic consensus* protocol. Such a transfer function can be viewed as a weight on the edge connecting node j to node i. As a special case, we consider graphs where these weights $Y_{ij}(s)$ are strictly-positive real transfer functions.

We now consider graphs formed by nodes with multiple terminals that are connected by weighted edges where the weights are represented as transfer functions taken from real rational functions analytic in the right half of the complex plane. These graphs are connected with the dynamic consensus protocol introduced earlier, and the dynamic Laplacian matrix

Copyright © AJSE Publishers

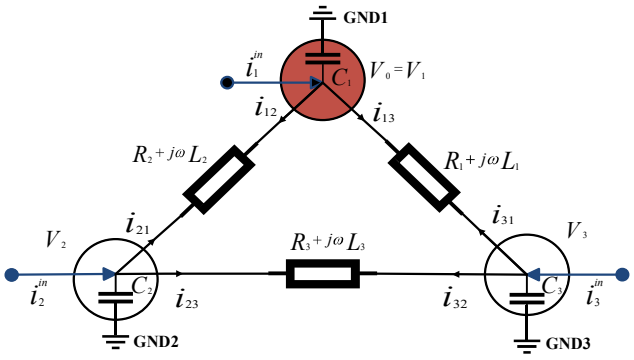


Fig. 2. Electrical network as dynamic graph.

is introduced. To illustrate how such a network can arise in applications, consider a simple electrical network shown in Fig. 2. This circuit can be represented as a dynamic graph by viewing the capacitors as nodes of the dynamic graph and the connections between capacitors as edges in which each edge is assigned a real rational transfer function $Z_{ij}(s)$ as its impedance. The node variables ($v_i, \forall i = 1, ..., N$) are voltages across the capacitors, and current is the flow along the edges between nodes. The weight of an edge can be viewed as the admittance of that edge, which we will denote $Y_{ij}(s) = \frac{1}{Z_{ij}(s)}$ for a given edge e_{ij}. If there is a potential difference $(v_i - v_j)$ between the nodes i and j, an electrical current i_{ij} will flow across the edge e_{ij} from i to j according to Ohm's law, which states that the potential drop across an edge e_{ij} must be equal to the product (in the complex frequency domain) of its impedance and the current flowing through it:

$$I_{ij}(s) = Y_{ij}(s)[V_i(s) - V_j(s)], \tag{17}$$

where $I_{ij}(s)$ and $V_i(s)$ are the Laplace transform of the current $i_{ij}(t)$ and the node voltage $v_i(t)$, respectively. Using Kirchhoff's law, it can be established that the algebraic sum of all of the currents entering a node i is equal to the algebraic sum of all currents leaving that node. Therefore, for Fig. 2, the dynamic model of each node is obtained as:

$$\begin{aligned} C\dot{v}_1(t) &= i_1^{in}(t) - (i_{12}(t) + i_{13}(t)); \\ C\dot{v}_2(t) &= i_2^{in}(t) - (i_{21}(t) + i_{23}(t)); \\ C\dot{v}_3(t) &= i_3^{in}(t) - (i_{31}(t) + i_{32}(t)). \end{aligned} \tag{18}$$

Noting that

$$i_{ij}(t) = -i_{ji}(t), \tag{19}$$

and, by combining (17) with the Laplace transformed version of (18) and setting $C = 1F$, we obtain:

$$\begin{aligned} V_i(s) =& \frac{1}{s} I_i^{in}(s) - \frac{1}{s} \sum_{j \in \mathcal{N}_i} Y_{ij}(s)[V_i(s) - V_j(s)] \\ &+ \frac{1}{s} v_i(0), \forall i = 1, 2, 3, \end{aligned} \tag{20}$$

where $\mathcal{N}_i$ is the set of neighbors node i. For the autonomous case ($I_i^{in}(s) = 0$), the dynamic model of each node with zero initial conditions ($v_i(0) = 0$) can then be written as:

$$V_i(s) = -\frac{1}{s} \sum_{j \in \mathcal{N}_i} Y_{ij}(s)[V_i(s) - V_j(s)], \forall i = 1, 2, 3. \tag{21}$$

Defining the node-potential vector $V(s) = [V_1(s), V_2(s), ..., V_N(s)]^T$, the overall system can be represented in the frequency domain by:

$$V(s) = -\frac{1}{s} L(s) V(s). \tag{22}$$

In the time domain, the above system can be represented by

$$\dot{v}(t) = -L(t) * v(t), \tag{23}$$

where $v(t) = [v_1(t), v_2(t), \ldots, v_N(t)]^T$ is the node potential vector and $L(t)$ In the autonomous case we have $V(s) = \frac{1}{s} L(s) V(s)$.

$$V(s) = (sI_N + L(s))^{-1} v(0), \tag{24}$$

In this expression the matrix $L(s) = [l_{ij}(s)]$ is given as:

$$l_{ij}(s) = \begin{cases} \sum_{j \in \mathcal{N}_i} Y_{ij}(s) & i = j \\ -Y_{ij}(s) & i \neq j \text{ and } (i,j) \in \mathcal{E} \\ 0 & \text{otherwise.} \end{cases} \tag{25}$$

This leads to:

$$L(s) = \begin{bmatrix} Y_{12}(s) + Y_{13}(s) & -Y_{12}(s) & -Y_{13}(s) \\ -Y_{12}(s) & Y_{12}(s) + Y_{23}(s) & -Y_{23}(s) \\ -Y_{13}(s) & -Y_{23}(s) & Y_{13}(s) + Y_{23}(s) \end{bmatrix}. \tag{26}$$

$L(s)$ is defined in this way as a dynamic Laplacian matrix of the dynamic graph G. In the sequel, we will also refer to the restriction $L(j\omega) = L(s)$ where $s = (0 + j\omega)$.

We can generalize this example by considering the general-undirected, dynamic consensus network shown in Fig. 3. For this system, the dynamic Laplacian has the form shown in (27). For this case, the dynamic Laplacian is a symmetric matrix ($\lambda_{ij}(s) = \lambda_{ji}(s)$ for all e_{ij}).

Based on the dynamics of a network's nodes and their topology, several consensus problems can be specified. This paper focuses on two types of dynamic consensus networks: directed and undirected. The dynamic consensus networks studied are:

- **Dynamic Network 1:** Directed dynamic networks with integrator nodes and dynamic edges:

$$\dot{x}_i(t) = -\sum_{j \in \mathcal{N}_i} \left[\lambda_{ij}^S(t) * x_i(t) - \lambda_{ij}^C(t) * x_j(t)\right],$$

or,

$$x_i(s) = -\frac{1}{s} \sum_{j \in \mathcal{N}_i} \left[\lambda_{ij}^S(s) x_i(s) - \lambda_{ij}^C(s) x_j(s)\right] \tag{28}$$

Copyright © AJSE Publishers

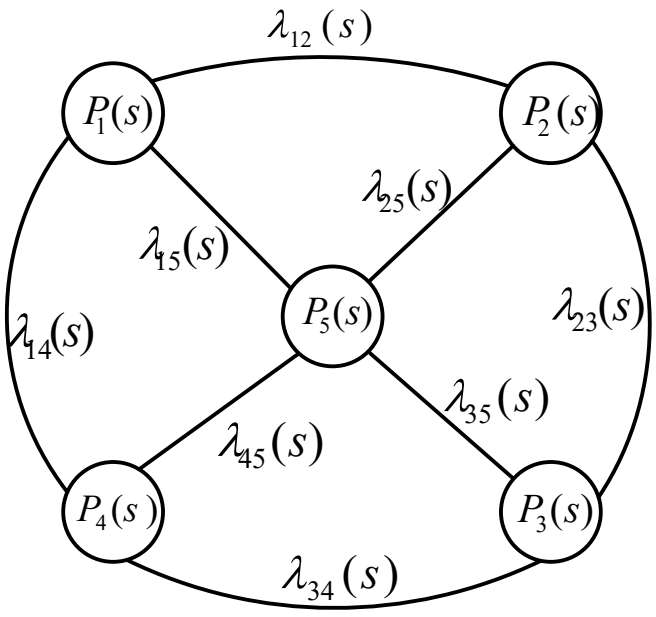


Fig. 3. Undirected, dynamic consensus network as undirected dynamic graph.

- **Dynamic Network 2:** Undirected dynamic networks with integrator nodes and strictly-positive-real (SPR) transfer function edges:

$$\dot{x_i}(t) = -\sum_{j\in\mathcal{N}_i}\left[\lambda_{ij}(t) * (x_i(t) - x_j(t))\right],$$

or,

$$x_i(s) = -\frac{1}{s}\sum_{j\in\mathcal{N}_i}\left[\lambda_{ij}(s)(x_i(s) - x_j(s))\right] \quad (29)$$

- **Dynamic Network 3:** Undirected dynamic networks with identical nodes and dynamic edges:

$$x_i(t) = -p(t) * \sum_{j\in\mathcal{N}_i}\left[\lambda_{ij}(t) * (x_i(t) - x_j(t))\right],$$

or,

$$x_i(s) = -p(s)\sum_{j\in\mathcal{N}_i}\left[\lambda_{ij}(s)(x_i(s) - x_j(s))\right] \quad (30)$$

- **Dynamic Network 4:** Undirected dynamic networks with heterogeneous nodes and dynamic edges:

$$x_i(t) = -p_i(t) * \sum_{j\in\mathcal{N}_i}\left[\lambda_{ij}(t) * (x_i(t) - x_j(t))\right],$$

$$x_i(s) = -p_i(s)\sum_{j\in\mathcal{N}_i}\left[\lambda_{ij}(s)(x_i(s) - x_j(s))\right] \quad (31)$$

Assumptions: For the dynamic networks (28 - 31), we make the following assumptions:

1) The node and edge processing in the proposed dynamic networks (28 - 31) are linear, time invariant LTI.
2) The dynamic topologies consist of dynamic edges $\lambda_{ij}(s)$ modeled as transfer functions. For the second proposed dynamic network (29), we assume the dynamics of the edges are strictly positive real (SPR) transfer functions.
3) The topology of a network can be directed or undirected. The first dynamic network (28) uses a directed topology, whereas the second dynamic network (29) uses an undirected topology.
4) Depending on the application, the flow is modeled differently. For instance, $[\lambda_{ij}^S(s)x_i(s) - \lambda_{ij}^C(s)x_j(s)]$ and $[\lambda_{ij}(s)(x_i(s) - x_j(s))]$ are two different ways of modeling flow, as is indicated by the previous work [22]. The difference between these two cases is illustrated in (28, 29). These flow models are rooted in the types of dynamic networks to be modeled per the motivation for each network. The first dynamic network (28) is based upon the motivation of modeling the thermal processes of buildings as directed dynamic graphs, while the second dynamic network (29) is based upon the motivation of modeling micro-grids of power systems as undirected dynamic graphs. This model is described in more detail in Section II.
5) The dynamics of the nodes can be integrators (28, 29) or more general dynamics (30, 31).
6) The dynamics of the nodes and of the edges can be identical (30) or heterogeneous (31).
7) These models are often autonomous, meaning that there are no input flows into the dynamic consensus networks. However, in some problems, we add inputs and disturbances into the general forms of the proposed dynamic consensus networks.

III. Dynamic Graphs Definitions

The previous work [22] showed how two different phenomena can be modeled in the form of a graph whose edges are transfer functions (i.e., dynamic systems). In this section, from the motivation of the models developed in the previous work, we generalize all the typical notations from (static) graph theory to the dynamic case. The development here parallels the notations in static graph.

Consider the example of a directed, dynamic graph shown in Fig. 4. Such graphs can be described as a set of nodes (or vertices) $\mathcal{N} = \{n_i\}$ connected by a set of edges $\mathcal{E}(s) = \{(n_i, n_j) : \ n_i, n_j \in \mathcal{N}\}$. Each edge is modeled as a transfer function $\lambda_{ij}(s)$. More formally, we can say that each edge $\lambda_{ij}(s) \in \mathcal{R}(s)$, where $\mathcal{R}(s)$ denotes the set of all complex-valued functions analytic in the open right-half complex plane (real rational functions). We assume there are no self-loops associated with any node. If there is an edge between nodes n_i and n_j, we say these nodes are adjacent (or neighbors). We denote the neighbors of node n_i as $\mathcal{N}_i = \{j : (n_i, n_j) \in \mathcal{E}(s)\}$. A path between two nodes is a sequence of edges by which it is possible to move along the sequence of arcs from one of the nodes to the other. If there is at least one node that has at least one path to every other node, the graph is said to be connected.

Later, we also view a node as implementing a transfer function that produces the node variable ($P_i(s)$ for $i = 1, 2, ..., N$, where N is number of nodes in the dynamic graph) by processing the incoming and outgoing flows.

As in the static case, the edges $e_{ij}(s)$ are ordered by the edge originating from node $P_i(s)$, known as the tail node, and

$$L(s) = \begin{bmatrix} \sum_{j=2,4,5} \lambda_{1j}(s) & -\lambda_{12}(s) & 0 & -\lambda_{14}(s) & -\lambda_{15}(s) \\ -\lambda_{12}(s) & \sum_{j=1,3,5} \lambda_{2j}(s) & -\lambda_{23}(s) & 0 & -\lambda_{25}(s) \\ 0 & -\lambda_{32}(s) & \sum_{j=2,4,5} \lambda_{3j}(s) & -\lambda_{34}(s) & -\lambda_{35}(s) \\ -\lambda_{41}(s) & 0 & -\lambda_{43}(s) & \sum_{j=1,3,5} \lambda_{4j}(s) & -\lambda_{45}(s) \\ -\lambda_{51}(s) & -\lambda_{52}(s) & -\lambda_{53}(s) & -\lambda_{54}(s) & \sum_{j=1,2,3,4} \lambda_{4j}(s) \end{bmatrix}. \quad (27)$$

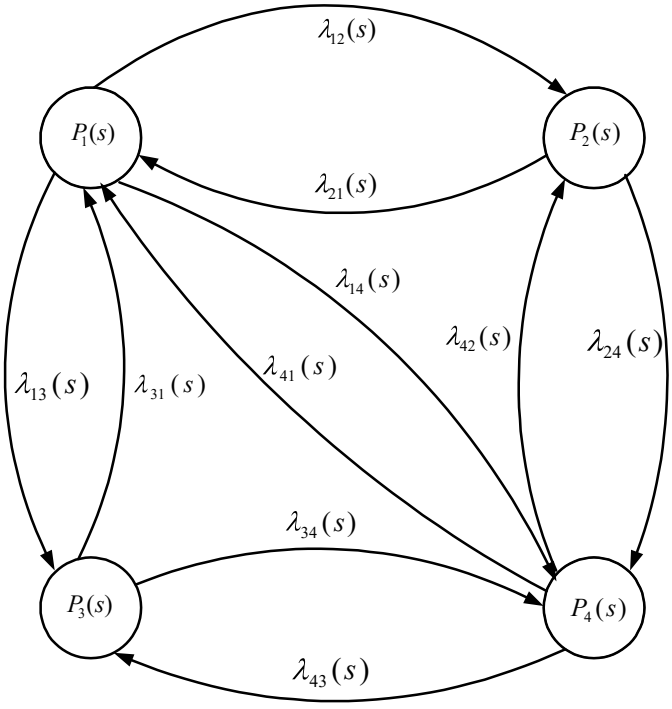


Fig. 4. Directed-Dynamic Graph.

terminating at node $P_j(s)$, known as the head node, which can be identified by the presence of an arrowhead. If the dynamics of the nodes are different, we call such dynamic networks heterogeneous. If the nodes have the same dynamics, the network is called homogeneous. A special case is when we have integrator nodes with dynamic edges.

Each node $P_i(s)$ in a directed dynamic graph, such as Fig. 4, is associated with a dynamic degree $\nu_i(s)$ representing the total sum of the dynamic edge weights that are connected to the node i. More specifically, each node has a dynamic in-degree $\nu_i^{in}(s)$ and a dynamic out-degree $\nu_i^{out}(s)$ representing the sum of the dynamic edge weights of the incoming and outgoing edges, respectively. Clearly $\nu_i(s) = \nu_i^{in}(s) + \nu_i^{out}(s)$. From these dynamic degree definitions, we can define three different dynamic degree matrices:

1) The dynamic in-degree matrix $D^{in}(s) = diag(\nu_i^{in}(s))$.
2) The dynamic out-degree matrix $D^{out}(s) = diag(\nu_i^{out}(s))$.
3) The dynamic degree matrix $D(s) = diag(\nu_i(s))$.

Notice that $D(s) = D^{in}(s) + D^{out}(s)$.

If a dynamic edge $e_{ij}(s)$ exists between two nodes i and j, these nodes are considered to be adjacent and are known as neighbors, and are denoted for a node $P_i(s)$ by $\mathcal{N}_i = \{j : (n_i, n_j) \in \mathcal{E}(s)\}$. As before, neighbors can be distinguished based upon whether they are associated with incoming or outgoing arcs. Thus, we can define three dynamic adjacency matrices:

1) The incoming dynamic adjacency matrix $A^{in}(s) = [a_{ij}^{in}(s)]$, is defined by

$$a_{ij}^{in}(s) = \begin{cases} \sum_{j \in \mathcal{N}_i} e_{ij}(s); (\ coming\ into\ n_i \text{ from } n_j) & \text{if } i \neq j \\ 0 & \text{otherwise.} \end{cases}$$

2) The dynamic outgoing adjacency matrix $A^{out}(s) = [a_{ij}^{out}(s)]$, is defined by

$$a_{ij}^{out}(s) = \begin{cases} \sum_{j \in \mathcal{N}_i} e_{ij}(s); (\ going\ out\ from\ n_i \text{ into } n_j) & \text{if } i \neq j \\ 0 & \text{otherwise.} \end{cases}$$

3) The dynamic adjacency matrix $A(s) = [a_{ij}(s)]$, is defined by

$$a_{ij}(s) = \begin{cases} \text{total} \sum_{\text{j} \in \mathcal{N}_\text{i}} \text{e}_\text{ij}(\text{s}); (\text{ between } n_i \text{ and } n_j) & \text{if } i \neq j \\ 0 & \text{otherwise.} \end{cases}$$

Notice that $A(s) = A^{in}(s) + A^{out}(s)$.

Another type of dynamic matrix is the dynamic incident matrix. For the incoming incident matrix, we will define two incident matrices: one indicates the direction of the edges connected to a node, where for node $P_i(s)$ the edge $e_{ij}(s)$ is given a value based upon being disconnected, incoming, or outgoing and denoted as B_S^{in} where "S" refers to static. Another matrix captures the transfer functions of the edges and is denoted as $B_D^{in}(s)$ where "D" refers to dynamic. Thus, the dynamic and static incoming incident matrices are defined as $B_D^{in}(s) = [b_{ij-D}^{in}(s)]$, $B_S^{in} = [b_{ij-S}^{in}]$, where,

$$b_{ij-D}^{in}(s) = \begin{cases} +\lambda_{ij}(s) & \text{if arc } j \text{ enters node } n_i \\ 0 & \text{otherwise.} \end{cases}$$

Copyright © AJSE Publishers

$$b^{in}_{ij-S} = \begin{cases} +1 & \text{if arc } j \text{ enters node } n_i \\ 0 & \text{otherwise.} \end{cases}$$

Similarly, we can define a dynamic and static outgoing incident matrices for a dynamic graph by, $B^{out}_D(s) = [b^{out}_{ij-D}(s)]$, $B^{out}_S = [b^{out}_{ij-S}]$, where,

$$b^{out}_{ij-D}(s) = \begin{cases} -\lambda_{ij}(s) & \text{if arc } j \text{ leaves node } n_i \\ 0 & \text{otherwise.} \end{cases}$$

$$b^{out}_{ij-S} = \begin{cases} -1 & \text{if arc } j \text{ leaves node } n_i \\ 0 & \text{otherwise.} \end{cases}$$

Also, we can define the dynamic and static incident matrices for a directed dynamic graph as $B_D(s) = [b_{ij-D}(s)]$, $B_S = [b_{ij-S}]$ where

$$b_{ij-D}(s) = \begin{cases} -\lambda_{ij}(s) & \text{if arc } j \text{ leaves node } n_i \\ +\lambda_{ij}(s) & \text{if arc } j \text{ enters node } n_i \\ 0 & \text{otherwise.} \end{cases}$$

$$b_{ij-S} = \begin{cases} -1 & \text{if arc } j \text{ leaves node } n_i \\ +1 & \text{if arc } j \text{ enters node } n_i \\ 0 & \text{otherwise.} \end{cases}$$

Notice that $B_D(s) = B^{in}_D(s) + B^{out}_D(s)$ and $B_S = B^{in}_S + B^{out}_S$.

Having derived dynamic degree, adjacency, and incident matrices, we can now give the dynamic equivalent of the static Laplacian matrix. The dynamic Laplacian matrix has spectral properties that indicate many facts about a graph. An undirected dynamic graph, one whose dynamic edges are not directionally-fixed, has a corresponding dynamic Laplacian matrix defined with no ambiguities by $L(s) = D(s) - A(s)$. More specifically, the dynamic Laplacian matrix is defined as $L(s) = [l_{ij}(s)]$, where

$$l_{ij}(s) = \begin{cases} \sum_{j \in \mathcal{N}_i} \lambda_{ij}(s) & i = j \\ -\lambda_{ij}(s) & i \neq j \text{ and } (i,j) \in \mathcal{E} \\ 0 & \text{otherwise} \end{cases}. \quad (32)$$

We can also define the dynamic Laplacian matrix of an undirected dynamic graph as

$$L(s) = BD(s)B^T = D(s) - A(s), \quad (33)$$

where $D(s) \in \mathbb{C}^{m \times m}$ is the dynamic degree matrix formed by the dynamic degree of the m edges, $B \in \mathbb{R}^{n \times m}$ is the static incident matrix that captures the orientations of the edges, and $A(s) \in \mathbb{C}^{n \times n}$ is the dynamic adjacency matrix.

In the static case, the definition of the Laplacian matrix for a directed graph requires adopting either an incoming or outgoing convention. Likewise, we define the dynamic Laplacian matrix utilizing the dynamic degree and dynamic adjacency matrices distinguishing between incoming and outgoing conventions. Examples include: $L^{in}(s) = D^{in}(s) - A^{in}(s)$ and $L^{out}(s) = D^{out}(s) - A^{out}(s)$. With these definitions, $L(s) = L^{in}(s) + L^{out}(s)$. Particular caution must be taken in noting that while $L = BB^T$ in the static case, $L(s) \neq B(s)B(s)^T$, $L^{in}(s) \neq B^{in}(s){B^{in}(s)}^T$ and $L^{out}(s) \neq B^{out}(s)B^{out}(s)^T$. To overcome this, in the sequel, we will use $L^{out}(s)$, where the outgoing dynamic Laplacian matrix can be defined using the incident matrices as follow:

$$L^{out}(s) = D^{out}(s) - A^{out}(s), \quad (34)$$

where $D^{out}(s)$ and $A^{out}(s)$ are the dynamic-outgoing degree and adjacency matrices, respectively. These matrices can be defined in a static case using the incident matrices as $D^{out} = B^{out}{B^{out}}^T$ and $A^{out} = -B^{out}{B^{in}}^T$. For the dynamic graphs, $D^{out}(s) \neq B^{out}(s)B^{out}(s)^T$ because the product of the dynamic incident matrices results in a matrix where its elements are the square of the edge dynamics (i.e., $\lambda_{ij}(s)^2$). To overcome this problem, we use the dynamic and static incident matrices (one will capture the edges orientation and the other will capture the edges dynamic) for defining the dynamic degree and adjacency matrices. Thus, the outgoing dynamic degree and adjacency matrices can be defined as:

$$\begin{aligned} D^{out}(s) &= B^{out}_D(s){B^{out}_S}^T, \\ A^{out}(s) &= -B^{out}_D(s){B^{in}_S}^T. \end{aligned} \quad (35)$$

By combining (34) and (35), the outgoing, dynamic Laplacian matrix $L^{out}(s)$ can be defined as

$$\begin{aligned} L^{out}(s) &= B^{out}_D(s){B^{out}_S}^T + B^{out}_D(s){B^{in}_S}^T \\ &= B^{out}_D(s)({B^{out}_S}^T + {B^{in}_S}^T) = B^{out}_D(s)B^T_S. \end{aligned} \quad (36)$$

A similar definition can be given for $L^{in}(s)$.

To illustrate, for the example shown in Fig. 4, the associated dynamic degree, adjacency, and Laplacian matrices are given by (37). Note that all $\lambda_{ij}(s)$ in (37) are transfer functions that describe the interconnections (edges) between the nodes. To understand the steady-state (consensus) properties of such a network, as well as its control, we are led to study the spectral properties of $L(s)$.

IV. Consensus Conditions of Networks with Integrator Nodes and Strictly Positive Real (SPR) Edges

In this section we show that the strictly positive realness of the edges is a sufficient condition for dynamic networks of the form (21) to reach a consensus (i.e., all node variables converge to the same constant value in the steady-state).

Analogous to the static consensus networks, we can define a dynamic matrix $L(s) = [L_{ij}(s)]$ to be a dynamic interconnection matrix as follows:

Definition 4.1: Define the set $\mathbb{S} = \{s : \text{Re}(s) \geq 0,\ s \neq 0\}$. $L(s)$ is a *dynamic interconnection matrix* if it satisfies the following properties:

1) $L(0)$ is an interconnnection matrix.
2) The elements of $L(s)$ have no poles in the closed right half of the complex plane.
3) The diagonal elements of $L(s)$ satisfy the following positivity condition: for all $s \in \mathbb{S}$, $\text{Re}\{L_{ii}(s)\} > 0$.

$$D^{out}(s) = \begin{bmatrix} \lambda_{12}(s)+\lambda_{13}(s)+\lambda_{14}(s) & 0 & 0 & 0 \\ 0 & \lambda_{21}(s)+\lambda_{24}(s) & 0 & 0 \\ 0 & 0 & \lambda_{31}(s)+\lambda_{34}(s) & 0 \\ 0 & 0 & 0 & \lambda_{41}(s)+\lambda_{42}(s)+\lambda_{43}(s) \end{bmatrix},$$

$$A^{out}(s) = \begin{bmatrix} 0 & \lambda_{12}(s) & \lambda_{13}(s) & \lambda_{14}(s) \\ \lambda_{21}(s) & 0 & 0 & \lambda_{24}(s) \\ \lambda_{31}(s) & 0 & 0 & \lambda_{34}(s) \\ \lambda_{41}(s) & \lambda_{42}(s) & \lambda_{43}(s) & 0 \end{bmatrix},$$

$$L^{out}(s) = \begin{bmatrix} \lambda_{12}(s)+\lambda_{13}+\lambda_{14}(s) & -\lambda_{12}(s) & -\lambda_{13}(s) & -\lambda_{14}(s) \\ -\lambda_{21}(s) & \lambda_{21}(s)+\lambda_{24}(s) & 0 & -\lambda_{24}(s) \\ -\lambda_{31}(s) & 0 & \lambda_{31}(s)+\lambda_{34}(s) & -\lambda_{34}(s) \\ -\lambda_{41}(s) & -\lambda_{42}(s) & -\lambda_{43}(s) & \lambda_{41}(s)+\lambda_{42}(s)+\lambda_{43}(s) \end{bmatrix}. \quad (37)$$

4) $L(s) = [L_{ij}(s)]$ satisfies the following diagonal dominance condition: for all $s \in \mathbb{S}$,

$$\mathrm{Re}\{L_{ii}(s)\} > \sum_{j \neq i} |L_{ij}(s)|.$$

Although condition 3 implies that $L_{ii}(s)$ is passive, the diagonal dominance condition does not require L_{ij}, $j \neq i$ to be passive.

Definition 4.2: A dynamic interconnection matrix is *connected* if $L(0)$ is connected.

In prior work [22], the diagonal elements of the dynamic matrix $L(s)$ of the proposed model are assumed to satisfy the following positivity condition (see Definition 4.1): for all $s \in \mathbb{S}$, $\mathrm{Re}\{L_{ii}(s)\} > 0$. Indeed, $L(s) = [L_{ij}(s)]$ in [22] satisfies the following diagonal dominance condition: for all $s \in \mathbb{S}$,

$$\mathrm{Re}\{L_{ii}(s)\} > \sum_{j \neq i} |L_{ij}(s)|.$$

Note that the dynamic Laplacian matrix $L(s)$ defined in this paper does not satisfy the properties of a dynamic interconnected matrix because the condition of diagonal dominance is not satisfied. Indeed, $L(s) = [L_{ij}(s)]$ in (25) satisfies the following equality:

$$L_{ii}(s) = \sum_{j \neq i, j=1}^{N} L_{ij}(s).$$

Furthermore, an approach to prove these last statements is based-on the Gershgorin circle theorem [11]) . Let $A \in \mathbb{C}^{n \times n}$ with elements a_{ij}. Let $R_i = \sum_{j \neq i} |a_{ij}|$ and define $D(a_{ii}, R_i)$ to be the closed disk centered at a_{ii} with radius R_i. Then every eigenvalue of A lies within at least one of the disks $D(a_{ii}, R_i)$.

The Gershgorin circle theorem Lemma A.1 is useful for analyzing a network satisfying the diagonal dominance condition as in [22], whereas it is not useful in application to the proposed dynamic network (21, 29).

In this paper, the edges are assumed to be strictly positive real. Thus, for the dynamic network (21) with strictly-positive real edges, we must look to the strictly positive realness and positive definiteness of the complex matrix. To proceed, we first define the concepts of strictly-positive real edges transfer functions and positive definiteness for complex matrices, and we give two definitions and two lemmas needed for our result:

Definition 4.3: [13] An $n \times n$ complex matrix A is called positive definite (PD) matrix if $Re[x^H Ax] > 0$ and positive semidefinite (PSD) if $Re[x^H Ax] \geq 0$, for all complex vector $x \in \mathbb{C}^n$, where x^H denotes the conjugate transpose of the vector x.

Definition 4.4: [28] An $n \times n$ proper rational transfer function matrix $G(s)$ is positive real if

1) Poles of all elements of $G(s)$ are in $\mathrm{Re}[s] \leq 0$.
2) For all real ω for which $j\omega$ is not a pole of any element of $G(s)$, the matrix $G(j\omega) + G^T(-j\omega)$ is positive semidefinite (PSD).
3) Any purely imaginary pole $j\omega$ of any element of $G(s)$ is a simple pole and the residue matrix $\lim_{s \to j\omega}(s - j\omega)G(s)$ is positive semidefinite Hermitian.

In addition, a real rational transfer function $h(s)$ is said to be strictly positive real (SPR) if there exists ϵ_0 such that $h(s-\epsilon)$ is positive real for $0 < \epsilon < \epsilon_0$.

Lemma 4.5: [13] A necessary and sufficient condition for a complex matrix A to be PD or PSD is that the Hermitian part $H(A) = \frac{1}{2}(A + A^H)$ must be PD or PSD, respectively.

Lemma 4.6: [13] Let $A \in \mathbb{C}^{n \times n}$. If $A + A^H$ is PD or PSD, then all the eigenvalues of matrix A have positive or non-negative real parts, respectively.

Using these definitions and lemmas, for the proposed dynamic network (21) we can give the following result:

Theorem 4.7: [14] Given system (22) for the dynamic consensus protocol (21) with edges that are SPR transfer functions, the following statements are true:

1) The dynamic Laplacian $L(s)$ is a complex, symmetric, positive semidefinite (CSPSD) matrix.
2) The real parts of the eigenvalues of $L(j\omega)$ are non-negative ($\mathrm{Re}[\lambda_i(L(j\omega))] \geq 0, \forall i \in 1, \ldots, n$), with $0 = \lambda_1(L(j\omega)) \leq \mathrm{Re}[\lambda_2(L(j\omega))] \leq ... \leq \mathrm{Re}[\lambda_n(L(j\omega))]$.
3) If $L(0)$ (the dynamic Laplacian $L(s)$ evaluated at $s = 0$) is the Laplacian matrix of a connected graph, then $s = 0$ is a distinct solution of $\det[sI_N + L(s)] = 0$ and all node variables will reach consensus under the protocol (21).

Proof:

1) The dynamic Laplacian is a CSPSD matrix:

From the definition of the dynamic Laplacian matrix (25), $Y_{ij}(j\omega) = Y_{ji}(j\omega)$ and then $L(j\omega)$ is a complex symmetric (CS) matrix, that is,

$$L(j\omega)^H = \bar{L}(j\omega)^T = \bar{L}(j\omega), \tag{38}$$

where $\bar{L}(j\omega)^T$ is the conjugate transpose of the dynamic Laplacian matrix. Using (38), the Hermitian part of the dynamic Laplacian matrix is

$$H(L(j\omega)) = \frac{1}{2}(L(j\omega) + L(j\omega)^H) = \text{Re}[L(j\omega)]. \tag{39}$$

From the above equation, we can conclude that if the dynamic Laplacian is a symmetric matrix, then the Hermitian part of $L(j\omega)$ is equal to the real part of the dynamic Laplacian matrix.

Based on Lemma 4.5, the necessary and sufficient condition for the dynamic Laplacian $L(j\omega)$ to be a PSD matrix is that the real part of $L(j\omega)$ be a PSD matrix. Based on (25), we can write $x^H\text{Re}[L(j\omega)]x = \sum_{j\in\mathcal{N}_i} \text{Re}[Y_{ij}(j\omega)]|x_i - x_j|^2 \geq 0$. Since $Y_{ij}(j\omega)$ is assumed to be strictly positive real, then $\text{Re}[Y_{ij}(j\omega)]$ is non-negative $\forall\omega \in \mathbb{R}$. Thus, for all $x \in C^n$, $x^H\text{Re}[L(j\omega)]x \geq 0$ and by Definition 4.3, $\text{Re}[L(j\omega)]$ is a PSD matrix. Therefore, the dynamic Laplacian matrix $L(j\omega)$ is also PSD matrix.

2) The real parts of the eigenvalues of $L(j\omega)$ are non-negative with $\lambda_1(L(j\omega)) = 0$:

We can show $\lambda_1(L(j\omega)) = 0$ with eigenvector $\mathbf{1}_N$ as follows: Based on the definition of $L(j\omega)$, we can observe that the rows of $L(j\omega)$ sum to zero, which implies that $L(j\omega)x = 0$. If all the entries of x are the same, then $\mathbf{1}_N$ is the eigenvector of eigenvalue 0.

Since $L(j\omega)$ is a PSD, then $Re[x^H L(j\omega)x] \geq 0$ for all $x \in \mathbb{C}^n$. In particular, it is true for $x = v_i$, where v_i is the ith eigenvector of $L(j\omega)$; thus, we have

$$\text{Re}[v_i^H L(j\omega)v_i] \geq 0. \tag{40}$$

Based on the definition of eigenvalues and eigenvectors $(L(j\omega)v_i = \lambda_i v_i)$, the above inequality can be written as:

$$\begin{aligned} &\text{Re}[v_i^H \lambda_i v_i] \geq 0 \\ \Rightarrow &\text{Re}[\lambda_i v_i^H v_i] \geq 0 \Rightarrow \text{Re}[\lambda_i \|v_i\|_2^2] \geq 0. \end{aligned} \tag{41}$$

Since $\|v_i\|_2^2 > 0$, then $\text{Re}[\lambda_i] \geq 0, \forall i \in 1, \ldots, n$.

Now, we can show that there is no purely imaginary eigenvalue of $L(j\omega)$: Suppose that there exists a non-zero pure imaginary eigenvalue $s = j\omega^*$ of $L(j\omega)$ with associated eigenvector v^*. Then we can write the following quadratic form:

$$v^{*^H} L(j\omega)v^* = \sum_{j\in\mathcal{N}_i} Y_{ij}(j\omega)|v_i^* - v_j^*|^2. \tag{42}$$

It follows from v being an eigenvector,

$$v^{*^H} L(j\omega)v^* = v^{*^H}(j\omega^* v^*) = j\omega^* \|v^*\|_2^2. \tag{43}$$

Combining (42) and (43) we get

$$j\omega^* \|v^*\|_2^2 = \sum_{j\in\mathcal{N}_i} Y_{ij}(j\omega^*)|v_i^* - v_j^*|^2, \tag{44}$$

By equating the real parts of the above equation, we can conclude that $v_i^* = v_j^*, \forall i, j$ because $Y_{ij}(j\omega^*)$ is strictly positive real. However, only eigenvalue with eigenvector $\mathbf{1}_N$ is 0, which is a contradiction.

3) To prove this part, it will be useful to define the set $\mathbb{S} = \{s : \text{Re}(s) \geq 0,\ s \neq 0\}$.

Since $L(0)$ is the Laplacian of a connected graph (see Definition 4.2), then it is obvious that $s = 0$ is a distinct solution of $\det[sI_n + L(s)] = 0$. We can show that all node variables will reach consensus if we can show that no solution to (22) exists with modes in $\mathbb{S}$. Let $\beta_i(s)$ be the least common multiple of the denominators of the ith row of $L(s)$, and let $\beta(s) = \text{diag}\begin{bmatrix}\beta_1(s) & \beta_2(s) & \cdots & \beta_m(s)\end{bmatrix}$. Let $\alpha(s) = \beta(s)L(s)$. Note that both $\beta(s)$ and $\alpha(s)$ are polynomial matrices. Then it can be shown (see e.g. [27]) that the solutions to (22) are given by $x(t)$ that satisfy

$$(s\beta(s) + \alpha(s))|_{s=\frac{d}{dt}}\, x(t) = 0.$$

By Lemma A.2, the allowable time domain modes of $x(t)$ are given by the roots of $\det(s\beta(s)+\alpha(s))$. Note that since $\beta(s)$ has no roots in the closed right half plane, the solutions to $\det(s\beta(s) + \alpha(s)) = 0$ on $\mathbb{S}$ are identical to the solutions of $\det\left(sI_n + \beta^{-1}(s)\alpha(s)\right) = \det(sI_n + L(s)) = 0$. By part 2, all the eigenvalues of $L(s)$ have non-negative real parts, which implies that all the roots of $\det[sI_n + L(s)] = 0$ have non-positive real-parts, so there are no solutions in $s \in \mathbb{S}$, which proves the claim. ■

As an example, consider the dynamic network of Fig. 2 with SPR transfer functions $Y_{ij}(s) = \frac{1}{Z_{ij}(s)}$ assigned for the edges, and for simplicity consider $Z_{ij} = R + j\omega L, R = 1\Omega$, and $L = 1H$. The dynamic Laplacian $L(s)$ can be written as

$$L(s) = \begin{bmatrix} \frac{2}{1+s} & -\frac{1}{1+s} & -\frac{1}{1+s} \\ -\frac{1}{1+s} & \frac{2}{1+s} & -\frac{1}{1+s} \\ -\frac{1}{1+s} & -\frac{1}{1+s} & \frac{2}{1+s} \end{bmatrix} \Rightarrow L(0) = \begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix}.$$

From the above, it is readily verified that $-L(0)$ is the Laplacian matrix of a connected graph. In fact, $L(0)$ is a connected interconnection matrix (see Definition 4.1). Applying Theorem 4.7, we expect that all node variables converge to consensus. This is verified in Fig. 5, which shows a simple simulation of the dynamic consensus protocol (21), with initial conditions for the node potentials taken to be $V(0) = [20, 60, 70]^T$.

V. Bounding the Eigenvalues of the Dynamic Laplacian Matrix

In this section, we develop a method to estimate the bounds of the eigenvalues of the dynamic Laplacian matrix $L(j\omega)$ for all ω. Our approach uses the idea of the *Dynamic Grounded Laplacian*. Motivated by the static case, we introduce the *Dynamic Grounded Laplacian* matrix to estimate lower and upper bounds for the real parts of the smallest and largest non-zero eigenvalues of the dynamic Laplacian matrix. We begin by defining the Dynamic Grounded Laplacian matrix. We then show how to obtain the lower and upper bounds for the eigenvalues of the dynamic Laplacian matrix.

Copyright © AJSE Publishers

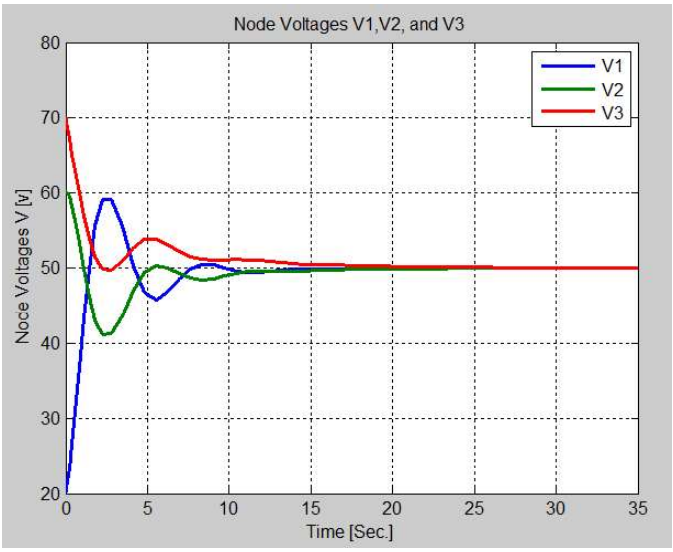


Fig. 5. Simulation of the node voltages of the dynamic network.

A. Dynamic Grounded Laplacian Matrix

The Dynamic Grounded Laplacian matrix can be defined from the dynamic Laplacian matrix by extending the definition of the Grounded Laplacian matrix in the case of static networks (e.g., see [2] and [3]) to the case of dynamic networks.

To illustrate the idea, we will define the Dynamic Grounded Laplacian matrix for the undirected dynamic graph that represents a simple electrical network as shown in Fig. 2. As we have mentioned earlier, this network can be modeled as a dynamic graph by representing the capacitors on the vertices as integrator nodes and the transmission lines or the branches between the nodes as SPR transfer functions edges.

Using Kirchhoff's law, it can be established that the dynamic Laplacian matrix $L(j\omega)$ expresses a mapping from input currents to node potentials as

$$\begin{bmatrix} I_1^{in}(j\omega) \\ I_2^{in}(j\omega) \\ I_3^{in}(j\omega) \end{bmatrix} = L(j\omega) \begin{bmatrix} V_1(j\omega) \\ V_2(j\omega) \\ V_3(j\omega) \end{bmatrix}, \tag{45}$$

where $L(j\omega)$ is defined by (26), where $Y_{ij}(j\omega)$ is the admittance between nodes i and j.

For a given dynamic network, assume that the node V_1 is connected to the ground (this node is colored red in the electrical network shown in Fig. 2). Grounding node V_1 means that the potential and the current of this node are zeros ($V_1(j\omega) = 0$ and $I_1^{in}(j\omega) = 0$).

Substituting $V_1(j\omega) = 0$ and $I_1^{in}(j\omega) = 0$ in the equation (45), we get

$$\begin{bmatrix} I_2^{in}(j\omega) \\ I_3^{in}(j\omega) \end{bmatrix} = \underbrace{\begin{bmatrix} Y_{12}(j\omega) + Y_{23}(j\omega) & -Y_{23}(j\omega) \\ -Y_{23}(j\omega) & Y_{13}(j\omega) + Y_{23}(j\omega) \end{bmatrix}}_{L_0(j\omega)} \begin{bmatrix} V_2(j\omega) \\ V_3(j\omega) \end{bmatrix}. \tag{46}$$

We call $L_0(j\omega)$, defined in this way, a Dynamic Grounded Laplacian matrix. We give this matrix this name because it is defined from the dynamic Laplacian by grounding the subset nodes $\mathbf{V_0} = V_1$ in this case.

These ideas can be generalized: Given a dynamic graph $\mathbf{G} = (\mathbf{V}, \mathbf{E})$, where $\mathbf{V}$ is a set of n nodes; $\mathbf{E} \subset \mathbf{V} \times \mathbf{V}$ is a set of m edges, and given a subset of grounded nodes $\mathbf{V_0} \subset \mathbf{V}$ consisting of $n_g < n$ nodes, $n_0 = n - n_g$, then the *Dynamic Grounded Laplacian* matrix $L_0(j\omega) \in \mathbb{C}^{n_0 \times n_0}$ is defined from the dynamic Laplacian matrix $L(j\omega)$ by removing all rows and columns corresponding to the nodes in $\mathbf{V_0}$. Note that this must be done in such a way that $L_0(j\omega)$ corresponds to a connected graph. The following lemma describes the properties of the dynamic Grounded Laplacian $L_0(j\omega)$.

Lemma 5.1: For an undirected, connected dynamic graph with SPR edges, the Dynamic Grounded Laplacian matrix $L_0(j\omega) \in \mathbb{C}^{n_0 \times n_0}$ is a complex, symmetric, positive definite (CSPD) matrix that is always invertible for all ω.

Proof:

Since the dynamic graph is undirected, the dynamic Laplacian matrix is symmetric $Y_{ij}(j\omega) = Y_{ji}(j\omega)$ and then $L_0(j\omega)$ is a complex symmetric (CS) matrix.

Now, we can show $L_0(j\omega)$ is PD by examining the positive definiteness of the Hermitian part of $L_0(j\omega)$ (see Definition 4.3),

$$\begin{aligned} H(L_0(j\omega)) &= \frac{1}{2}(L_0(j\omega) + L_0(j\omega)^H) \\ &= \frac{1}{2}(L_0(j\omega) + \bar{L}_0(j\omega)) = \mathrm{Re}[L_0(j\omega)]. \end{aligned}$$

Based on Lemma 4.5, the positive definiteness of $L_0(j\omega)$ is due to $x^T \mathrm{Re}[L_0(j\omega)]x = \sum_{j \in \mathcal{N}_i^0} \mathrm{Re}[Y_{ij}(j\omega)]|x_i - x_j|^2 > 0$ for all $x \in \mathbb{R}^{n_0}$, where $\mathcal{N}_i^0$ is the set of neighborhood nodes to the node i in the remaining graph after grounding. Thus, the Dynamic Grounded Laplacian matrix $L_0(j\omega)$ is always invertible for all ω. ■

B. Finding Bounds for the Eigenvalues of the Dynamic Laplacian Matrix

In this subsection, the bounds for the eigenvalues of the dynamic Laplacian are estimated using the properties of the dynamic Laplacian and Dynamic Grounded Laplacian matrices.

1) Lower Bound for the Smallest Non-Zero Eigenvalue of the Dynamic Laplacian Matrix: The following theorem provides a lower bound of the dynamic Laplacian matrix. Later this bound will be used to characterize the stability of the closed loop system in the case of distributed controllers.

Theorem 5.2: Let $L(j\omega) \in \mathbb{C}^{n \times n}$ be the dynamic Laplacian matrix for an undirected dynamic graph with SPR edges. Let $\lambda_i(L(j\omega))$ and $\lambda_i(L_0(j\omega))$ denote the eigenvalues of $L(j\omega)$ and $L_0(j\omega)$, respectively, for any choice of a boundary set of grounded nodes **V0**, for which the graph is connected. Assume that $\mathrm{Re}[\lambda_1(L(j\omega))] \leq \mathrm{Re}[\lambda_2(L(j\omega))] \leq \ldots \leq \mathrm{Re}[\lambda_n(L(j\omega))]$ and $\mathrm{Re}[\lambda_1(L_0(j\omega))] \leq \mathrm{Re}[\lambda_2(L_0(j\omega))] \leq \ldots \leq \mathrm{Re}[\lambda_{n_0}(L_0(j\omega))]$, $n_0 < n$. Then a lower bound for the real part of the smallest non-zero eigenvalue of the dynamic Laplacian is given by

$$\min_{\omega} \mathrm{Re}[\lambda_2(L(j\omega))] \geq \frac{1}{\|\mathrm{trace}((\mathrm{Re}[L_0(j\omega)])^{-1})\|_\infty}. \tag{47}$$

Copyright © AJSE Publishers

Proof:

The proof of this Theorem is given in the following steps:

1) We begin the proof by finding lower bounds for the eigenvalues of the Hermitian part of a Dynamic Grounded Laplacian matrix $\lambda_i(H(L_0(j\omega)))$:
Based on Lemma A.3, these bounds can be obtained as

$$\lambda_i(H(L_0(j\omega))) \geq \frac{1}{\text{trace}((\text{Re}[L_0(j\omega)])^{-1})}, \quad (48)$$

for all $\omega, i \in 1, 2, ..., n$.

2) In this step, we derive the inequality that relates the eigenvalues of the matrices $H(L_0(j\omega))$ and $H(L(j\omega))$: Applying the interlacing eigenvalue Theorem A.4, we can conclude that the eigenvalues of $H(L(j\omega)) \in \mathbb{R}^{n\times n}$ and $H(L_0(j\omega)) \in \mathbb{R}^{n_0\times n_0}$ are interlaced for $i = 1, 2, ..., n_0$, as

$$\begin{aligned}\lambda_i(H(L(j\omega))) &\leq \lambda_i(H(L_0(j\omega))) \\ &\leq \lambda_{i+(n-n_0)}(H(L(j\omega))). \end{aligned} \quad (49)$$

Combining (48) and (49), we get

$$\lambda_i(H(L(j\omega))) \geq \frac{1}{\text{trace}((\text{Re}[L_0(j\omega)])^{-1})}, \quad (50)$$

for all ω and $i = (n - n_0) + 1, ..., n$.

3) Here, we derive the relationship between the real part of the smallest and largest nonzero eigenvalues of $L(j\omega)$ and $H(L(j\omega))$. This relationship can be achieved using Theorem A.5 as

$$\sum_{i=n-k}^{n} \text{Re}[\lambda_i(L(j\omega))] \leq \sum_{i=n-k}^{n} \lambda_i(H(L(j\omega))), \quad (51)$$

$k = 0, 1, ..., n - 1$, with equality for $k = n - 1$.
Based on Mirsky Theorem A.5, the relationship between the eigenvalues of the matrices $L(j\omega)$ and $H(L(j\omega))$ can be obtained by substituting $k = n - 3$, $k = n - 1$, and $k = 0$ in (51) and then combining the result. Thus, the inequalities that characterize the relationship between the eigenvalues of these matrices are given by Lemma A.6 as

$$\text{Re}[\lambda_2(L(j\omega))] \geq \lambda_2(H(L(j\omega))) \quad (52)$$

and

$$\text{Re}[\lambda_n(L(j\omega))] \leq \lambda_n(H(L(j\omega))), \quad (53)$$

where $\text{Re}[\lambda_2(L(j\omega))]$ and $\text{Re}[\lambda_n(L(j\omega))]$ are the real part of the smallest and largest nonzero eigenvalues of $L(j\omega)$ and $H(L(j\omega))$, respectively. Note that these result are obtained for the case of a connected dynamic graph with SPR edges, which is mean $\text{Re}[\lambda_1(L(j\omega))] = \lambda_1(H(L(j\omega))) = 0$.

4) Finally, a lower bound for the real part of the smallest non-zero eigenvalue of $L(j\omega)$ can be obtained by combining (50) and (52) as

$$\text{Re}[\lambda_2(L(j\omega))] \geq \frac{1}{\text{trace}((\text{Re}[L_0(j\omega)])^{-1})}, \forall\omega. \quad (54)$$

Taking the minimum values of both sides in the above inequality over all ω, yields

$$\min_\omega \text{Re}[\lambda_2(L(j\omega))] \geq \min_\omega [\frac{1}{\text{trace}((\text{Re}[L_0(j\omega)])^{-1})}];$$

$$\min_\omega \text{Re}[\lambda_2(L(j\omega))] \geq \frac{1}{\max_\omega[\text{trace}((\text{Re}[L_0(j\omega)])^{-1})]}. \quad (55)$$

Therefore, a lower bound for the real part of the smallest non-zero eigenvalue of the dynamic Laplacian matrix is given by

$$\min_\omega \text{Re}[\lambda_2(L(j\omega))] \geq \frac{1}{\|\text{trace}((\text{Re}[L_0(j\omega)])^{-1})\|_\infty}. \quad (56)$$

■

2) An Upper Bound for the Largest Eigenvalue of the Dynamic Laplacian Matrix: The following theorem provides an upper bound for the real part of the largest eigenvalue of the dynamic Laplacian matrix:

Theorem 5.3: Let $L(j\omega) \in \mathbb{C}^{n\times n}$ be the dynamic Laplacian matrix for an undirected dynamic graph with SPR edges. Let $\lambda_i(L(j\omega))$ denote the eigenvalues of $L(j\omega)$, ordered so that $\text{Re}[\lambda_1(L(j\omega))] \leq \text{Re}[\lambda_2(L(j\omega))] \leq ... \leq \text{Re}[\lambda_n(L(j\omega))]$. Then an upper bound for the real part of the largest eigenvalue of the dynamic Laplacian matrix is given by

$$\max_\omega \text{Re}[\lambda_n(L(j\omega))] \leq 2 \max_i \|\text{Re}[D(j\omega)(i,i)]\|_\infty, \quad (57)$$

where $D(j\omega)(i,i)$ is the ith diagonal element of the dynamic Degree matrix of the dynamic graph.

Proof:

First, we show that the $H[2D(j\omega) - L(j\omega)]$ is a PSD matrix for all ω. Because of the positivity realness of the edges and the symmetry property (because the assumed dynamic graph uses undirected topology), we can write

$$\begin{aligned} &H[2D(j\omega) - L(j\omega)] = H[2D(j\omega) - (D(j\omega) - A(j\omega)] \\ &= H[D(j\omega) + A(j\omega)] = \text{Re}[D(j\omega) + A(j\omega)], \end{aligned} \quad (58)$$

where $D(j\omega)$ and $A(j\omega)$ are the dynamic degree and adjacency matrix of the dynamic graph. In addition, the symmetry property of the matrix $[D(j\omega) + A(j\omega)]$ can be easily verified from the following definition: $[D(j\omega) + A(j\omega)] = [l_{ij}(j\omega)]$, where

$$l_{ij}(j\omega) = \begin{cases} \sum_{j\in\mathcal{N}_i} Y_{ij}(j\omega) & i = j \\ +Y_{ij}(j\omega) & i \neq j \text{ and } (i,j) \in \mathcal{E} \\ 0 & \text{otherwise.} \end{cases} \quad (59)$$

Note that the matrix defined above is different from the dynamic Laplacian matrix $L(s)$. The difference is the off-diagonal terms are positive.

By Lemma 4.5, the necessary and sufficient condition for the matrix $H(2D(j\omega) - L(j\omega)) = \text{Re}[D(j\omega) + A(j\omega)]$ to be a PSD matrix is that the real part of $[D(j\omega) + A(j\omega)]$ be a PSD matrix.

Copyright © AJSE Publishers

Based on Definition 4.3 and (59), we can write

$$x^T \text{Re}[D(j\omega) + A(j\omega)]x = \sum_{j \in \mathcal{N}_i} \text{Re}[Y_{ji}(j\omega)](x_i + x_j)^2 \geq 0,$$

for all vectors $x \in \mathbb{R}^n$. Because of the strictly positive realness of the edges, then $\text{Re}[Y_{ji}(j\omega)]$ is non-negative $\forall \omega \in \mathbb{R}$. Thus, by Definition 4.3, $\text{Re}[D(j\omega) + A(j\omega)]$ is a PSD matrix. Therefore, the matrix $H[2D(j\omega) - L(j\omega)]$ is also a PSD matrix. Hence,

$$H[2D(j\omega) - L(j\omega)] \geq 0 \Leftrightarrow H(L(j\omega)) \leq 2H(D(j\omega)). \tag{60}$$

From the above inequality, we can write

$$\lambda_i(H(L(j\omega))) \leq 2\lambda_i(H(D(j\omega))) = 2\text{Re}[D(j\omega)(i,i)]. \tag{61}$$

Thus, the largest eigenvalue of the Hermitian part of the dynamic Laplacian can be upper-bounded by

$$\begin{aligned} &\lambda_n(H(L(j\omega))) \leq 2 \max_i (\max_\omega \text{Re}[D(j\omega)(i,i)]) \\ \Leftrightarrow \quad &\lambda_n(H(L(j\omega))) \leq 2 \max_i \|\text{Re}[D(j\omega)(i,i)]\|_\infty . \end{aligned} \tag{62}$$

Combining (53) and (62), we get

$$\text{Re}[\lambda_n(L(j\omega))] \leq 2 \max_i \|\text{Re}[D(j\omega)(i,i)]\|_\infty . \tag{63}$$

Thus,

$$\max_\omega \text{Re}[\lambda_n(L(j\omega))] \leq 2 \max_i \|\text{Re}[D(j\omega)(i,i)]\|_\infty .$$

■

The lower and upper bounds for the eigenvalues of the dynamic Laplacian matrix are useful for stability analysis of dynamic consensus networks under feedback control. In the next section we show how these bound can be used to establish the stability of an undirected dynamic consensus network under distributed control.

VI. Stability Analysis of Dynamic Consensus Networks with Distributed Controllers

In this section, we provide conditions for the stability of undirected dynamic networks controlled using distributed controllers. The bounds of the eigenvalues of the dynamic Laplacian matrix $\lambda_i(L(j\omega))$, which can be found using the methodology from the previous section, can be used to analyze the stability of such dynamic networks using the Nyquist graphical-stability test [12]. The authors of [7] and [12] applied tools from graph theory to relate the topology of a static network to the stability of the overall system. They also provided a Nyquist criterion, which uses the eigenvalues of the static Laplacian matrix to determine the effect of the static graph on stability. Here, we extend this stability criterion from static networks to dynamic networks. Before analyzing the stability of the dynamic consensus networks, we will formulate our problem for the proposed dynamic network (undirected dynamic consensus network with identical LTI nodes and dynamic edges).

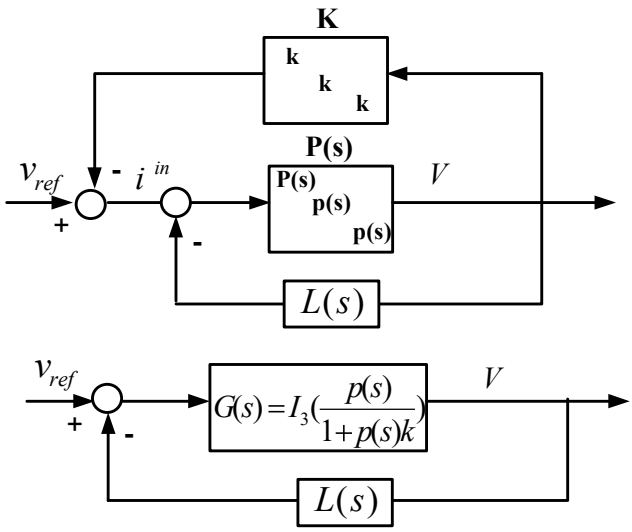


Fig. 6. Upper figure: Block diagram of the controlled system. Lower figure: Equivalent standard feedback diagram.

A. Problem Formulation

Consider a dynamic network with N identical nodes and with dynamic edges e_{ij} (here, we assume the dynamics of each edge is a SPR transfer function), as in Fig. 2, where the voltage of the ith node is represented as $v_i, i = 1, \ldots, N$. Furthermore, we assume that the nodes have identical dynamics $p(s)$ and controllers $k(s)$.

Consider the dynamic network Fig. 2 as defined in (20) and let the integrator nodes be replaced by a common transfer function such as $p_i(s) = p(s)$. For example, assume that the integrator nodes can be replaced by filters with a common transfer function $p_i(s) = p(s) = \frac{\beta}{s+\alpha}$. Thus, the dynamics of each node can be rewritten in the time domain as

$$\frac{1}{\beta}(\dot{v}_i(t) + \alpha v_i(t)) = i_i^{in}(t) - \sum_{j \in \mathcal{N}_i} y_{ij}(t) * (v_i(t) - v_j(t)), \tag{64}$$

$\forall i = 1, \ldots, N$. Now suppose the node voltage v_i of each node is controlled by a static controller k, that determines the input $i_i^{in}(t)$. Then the overall system can be expressed in the frequency domain as

$$\begin{aligned} P(s)^{-1}V(s) &= I^{in}(s) - L(s)V(s) \\ I^{in}(s) &= v^{ref} - KV(s), \end{aligned} \tag{65}$$

where $P(s) = \text{diag}(p(s), p(s), \ldots p(s))$, $K = \text{diag}(k(s), k(s), \ldots k(s))$, where $k(s) = k$ and $V(s)$, $I^{in}(s)$ are the Laplace transform of the vectors $v(t)$ and $i^{in}(t)$, respectively. The control scheme of this problem can be depicted as shown in the upper part of Fig. 6 (for clarity, shown for $N = 3$). We will apply the Nyquist criterion for this block diagram for our controller design, using the estimated bounds of the eigenvalues of the dynamic Laplacian matrix. Thus, the upper block diagram is first transformed into an equivalent standard feedback diagram as shown in the part of Fig. 6. Note that the identical controller $k(s) = k$ in the equivalent standard feedback diagram is located in the forward path ($G(s)$).

Copyright © AJSE Publishers

The closed-loop transfer function $H(s)$ from $v^{ref}(s)$ to $V(s)$ can be expressed as:

$$H(s) = G(s)(I_3 + G(s)L(s))^{-1}, \tag{66}$$

where $G(s) = I_N \frac{p(s)}{1+p(s)k}$.

The feedback matrix $(L(s))$ in the equivalent standard feedback diagram of the controlled dynamic network in the lower part of Fig. 6 is a dynamic matrix, whereas this matrix is a constant in the static case. This leads us to proceed with bounding the eigenvalues of the dynamic Laplacian matrix in order to locate them for analyzing the stability of the closed loop system using the Nyquist criterion. These bounds can be predicted for static networks based on properties of the graph [7]. The authors of [7]] considered locations of the eigenvalues of the Laplacian matrix to investigate the stability of a consensus network. The region of reciprocal eigenvalues of the Laplacian matrix can be defined using the estimated regions of the smallest and largest real parts of the non-zero eigenvalues of the dynamic Laplacian matrix as introduced in the previous section. Using these estimated regions, the stability conditions of the closed loop system $H(j\omega)$ can be analyzed with respect to the eigenvalues of $L(j\omega)$ $(\lambda_i(L(j\omega)))$ as we will see in the next section.

B. Stability Analysis

Here we analyze the stability of the closed-loop system $H(s)$ with respect to the eigenvalues of $L(j\omega)$ $(\lambda_i(L(j\omega)))$. Using Nyquist's graphical-stability test, the stability of the closed loop system (66) with identical node dynamics $p(s)$ and controller $k(s) = k$ can be analyzed as follows:

By Theorem 2 in [10], it can be seen that closed loop system $H(j\omega)$ in (66) is stable if and only if the Nyquist plot of

$$\prod_{i=1}^{N}[1 + \lambda_i(L(j\omega))G(j\omega)] \tag{67}$$

makes m c.c.w encirclements of the origin, where m is the number of unstable poles of $G(j\omega)L(j\omega)$, and

$$G(j\omega) = I_N \frac{p(j\omega)}{1 + p(j\omega)k}.$$

Now consider the case when $G(j\omega)$ and $L(j\omega)$ are stable. Then the stability analysis condition stated above is equivalent to the closed loop system (66) is stable if the Nyquist plot of

$$\prod_{i=1}^{N}[1 + \lambda_i(L(j\omega))G(j\omega)]$$

makes zero encirclements around the origin, which is equivalent to the Nyquist plot of

$$\lambda_i(L(j\omega))G(j\omega)$$

makes zero encirclements around the $-1 + 0j$, for all i.

Furthermore, the above stability condition is equivalent to the closed loop system $H(j\omega)$ is stable if the Nyquist plot of a stable transfer function $G(j\omega)$ does not encircle the points $\frac{-1}{\lambda_i(L(j\omega))}$ for all ω, and $i = 2, \dots n$ (see [12] in the case of static Laplacian matrix). Thus, satisfying this statement means that the closed loop system $H(j\omega)$ is stable and then all the eigenvalues of $H(j\omega)$ are in the left half of the complex plane. Note that the stability of the system does not mean the system will achieve consensus. Thus, we require an additional condition for guaranteeing consensus, which is the connectivity of the graph that is associated with the static Laplacian matrix $L(0)$, because under the connectivity condition, the zero eigenvalue of the dynamic Laplacian is distinct and then we can guarantee consensus.

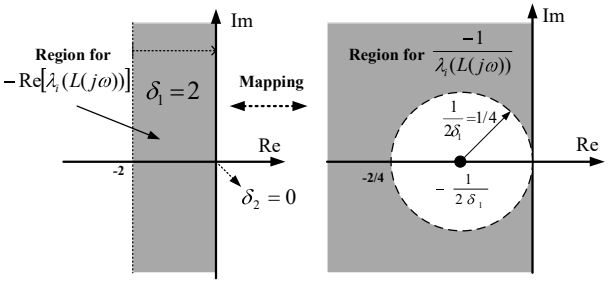


Fig. 7. Left figure: Inclusion region of $-\mathrm{Re}[\lambda_i(L(j\omega))]$. Right figure: Inclusion region of $-\frac{1}{\lambda_i(L(j\omega))}$.

In order to apply the above conditions to designing a controller $k(s)$ such that the closed loop system $H(s)$ is stable, the bounds of the eigenvalues of the dynamic Laplacian must be estimated. These bounds are estimated in Section V-B and given by (47) and (57).

C. Illustrative Simulations

To illustrate how our approach can be used for the design of controllers, consider the dynamic network Fig. 2 with the dynamic Laplacian $L(s)$ defined by (26) with parameters $Y_{12}(s) = \frac{1}{4s+4}$ and $Y_{13} = Y_{23}(s) = \frac{1}{2s+2}$. Using these parameters, the bounds for the real parts of all non-zero eigenvalues of $L(j\omega)$, $-\mathrm{Re}[\lambda_i(L(j\omega))]$ can be estimated by (47) and (57) as

$$\begin{aligned}&[-\max_{\omega}\mathrm{Re}[\lambda_n(L(j\omega)), -\min_{\omega}\mathrm{Re}[\lambda_2(L(j\omega))]] \subset \\ &[-2\max_{i}\|\mathrm{Re}[D(j\omega)(i,i)]\|_\infty, \frac{-1}{\|\mathrm{trace}((\mathrm{Re}[L_0(j\omega)])^{-1})\|_\infty}] \\ &= [-2, 0].\end{aligned} \tag{68}$$

Using the above estimated bounds, the region containing negative of the eigenvalues of the dynamic Laplacian matrix $L(j\omega)$ indicated as the shaded region on the left side of Fig. 7. For the stability analysis described above, we need to define the region of $\frac{-1}{\lambda_i(L(j\omega))}$ for all ω, $i = 1 \dots N$. Therefore, the bound $-\max_\omega \mathrm{Re}[\lambda_3(L(j\omega)) \geq -4$ is mapped to the exterior of a circle centered at $(-1/4, 0)$ with radius $1/4$. Furthermore, the bound $-\min_\omega \mathrm{Re}[\lambda_2(H(j\omega)) \leq 0]$ is mapped to the left half of the complex plane. Thus, the region of $\frac{-1}{\lambda_i(L(j\omega))}$ is the intersection between the two mapped regions, which is indicated as the shaded region on the right side of Fig. 7. That is a conservative estimate; all the points $-\frac{1}{\lambda_i(L(j\omega))}$ lie in this shaded region. Thus if the Nyquist plot of $G(j\omega)$ does not enter this region, we can be sure consensus occurs.

For this example, assume the nodes of the given dynamic network have a common stable transfer function $p(s) = \frac{\beta}{s+\alpha}$,

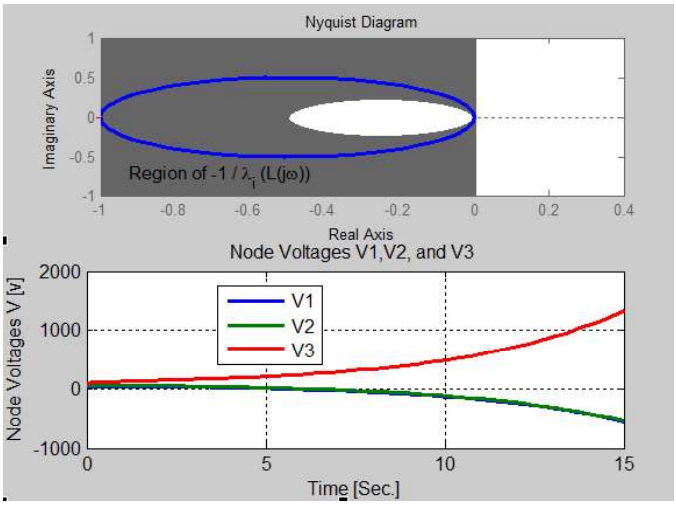


Fig. 8. Upper figure: Nyquist plot of $G(j\omega)$ (blue curve) with the region of $\frac{-1}{\lambda_i(L(j\omega)}$ (shaded region). Lower figure: Simulation of the node voltages in the dynamic network.

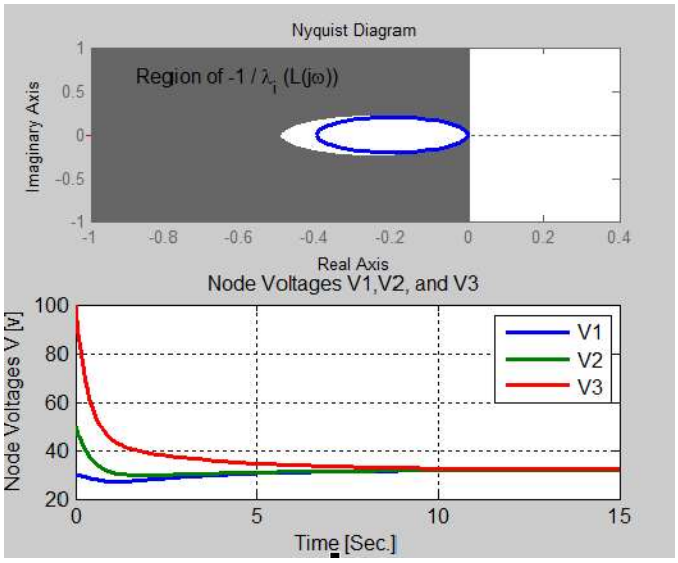


Fig. 9. Upper figure: Nyquist plot of $G(j\omega)$ (blue curve) with the region of $\frac{-1}{\lambda_i(L(j\omega)}$ (shaded region). Lower figure: Simulation of the node voltages in the dynamic network.

$\beta = -1, \alpha = 2$. Further, assume a simple static controller $k(s) = k$. Then the stability of the closed loop system discussed above and the estimated bounds can be verified using the Nyquist plot of $G(j\omega)$ with respect to the defined region of $\frac{-1}{\lambda_i(L(j\omega))}$ by considering the following two cases:

Case 1: With a common static controller $k = 1$, the Nyquist plot of $G(j\omega) = I_3 \frac{p(j\omega)}{1+p(j\omega)k}$ (blue curve as shown in the upper side of Fig. 8) does encircle (i.e., intersects) one of the points $\frac{-1}{\lambda_i(L(j\omega))}$. Thus, the closed loop system $H(j\omega)$ in (66) is unstable as shown in the lower side of Fig. 8.

Case 2: If we change the static controller to $k = -0.5$, then the Nyquist plot of $G(j\omega) = I_3 \frac{p(j\omega)}{1+p(j\omega)k}$ (blue curve as shown in the upper side of Fig. 9) does not encircle any point $\frac{-1}{\lambda_i(L(j\omega))}$ because it does not intersect the shaded region. Thus, the closed loop system is stable as shown in the lower side of Fig. 9.

Thus, we can conclude that $k(s)$ stabilizes the closed loop system $H(j\omega)$ if the number of net encirclements of the points $\frac{-1}{\lambda_i(L(j\omega))}$ by the Nyquist plot of the stable transfer function $G(j\omega)$ is zero for all nonzero $\lambda_i(H(j\omega)$. In other words, the stability of the closed-loop system $H(j\omega)$ can be investigated from the locations of the eigenvalues of the dynamic Laplacian matrix $L(j\omega)$, which was estimated in the previous Section, in relation to a Nyquist plot of stable $G(j\omega)$.

In order to design a controller for the proposed dynamic network (identical LTI systems with dynamic edges) that causes all states of nodes to achieve consensus, the controller $k(s)$ must be chosen such that the closed-loop system $H(j\omega)$ is stable or, ideally, reaches consensus. In other words, we must avoid the net encirclement of $-\frac{1}{\lambda_i(L(s))}$ by the Nyquist plot of the stable transfer function $G(j\omega)$ for all nonzero, as was in the second case. For consensus investigation, we require an additional condition, which is the connectedness of $L(0)$ as was previously mentioned, meaning that the zero eigenvalue is distinct.

VII. Conclusion

Motivated by several physical examples, including modeling micro-grids of power systems as undirected dynamic consensus network that we presented, we considered dynamic networks whose nodes are integrators and whose edges are SPR transfer functions representing dynamical systems that couple the nodes. We presented a framework for dynamic graphs and dynamic consensus networks. This framework introduced the idea of dynamic degree, adjacency, incident, and Laplacian matrices in a way that naturally extends these concepts from the static case. From this, equivalent concepts for dynamic interconnection matrices and dynamic consensus networks were defined. We showed that the strictly positive realness of the edges is a sufficient condition for these kinds of dynamic networks to reach a consensus. We developed a method to estimate the bounds of the eigenvalues of the dynamic Laplacian matrix. Our approach used the idea of the *Dynamic Grounded Laplacian*. These bounds were used for a Nyquist stability analysis of a closed loop system comprising an undirected dynamic network subject to distributed control. The region of reciprocal eigenvalues of the Laplacian matrix was defined using the prediction regions of the smallest and largest real parts of the non-zero eigenvalues of the dynamic Laplacian matrix. Using these prediction-regions, the stability conditions of the closed loop system was analyzed with respect to the eigenvalues of the dynamic Laplacian matrix. In future work, we extend these notions to consider the stability and consensus for a dynamic consensus networks with non-identical nodes and dynamic edges.

Acknowledgments

The authors would like to thank Professor Kevin L. Moore and Professor Tyrone L. Vincent at Colorado School of Mines for thier helps and advices.

Copyright © AJSE Publishers

REFERENCES

[1] H. Ando, Y. Oasa, I. Suzuki, and M. Yamashita. Distributed memoryless point convergence algorithm for mobile robots with limited visibility. *IEEE Transactions on Robotics and Automation*, 15(5):818–828, 1999.

[2] P. Barooah and J.P. Hespanha. Graph effective resistance and distributed control: Spectral properties and applications. In *Proceedings of 45th IEEE Conference on Decision and Control*, pages 3479 –3485, dec. 2006.

[3] P. Barooah and J.P. Hespanha. Estimation on graphs from relative measurements. *IEEE Control Systems*, 27(4):57 –74, aug. 2007.

[4] J. Cortes, S. Martinez, T. Karatas, and F. Bullo. Coverage control for mobile sensing networks. *IEEE Transactions on Robotics and Automation*, 20(2):243–255, 2004.

[5] M.H. DeGroot. Reaching a consensus. *Journal of the American Statistical Association*, pages 118–121, 1974.

[6] F. Dorfler and F. Bullo. Synchronization and transient stability in power networks and non-uniform kuramoto oscillators. In *Proceedings of 2010 American Control Conference*, pages 930–937. IEEE, 2010.

[7] J Alexander Fax and Richard M Murray. Information flow and cooperative control of vehicle formations. *IEEE transactions on automatic control*, 49(9):1465–1476, 2004.

[8] J.A. Fax and R.M. Murray. Information flow and cooperative control of vehicle formations. *IEEE Transactions on Automatic Control*, 49(9):1465–1476, 2004.

[9] R.A. Freeman, P. Yang, and K.M. Lynch. Distributed estimation and control of swarm formation statistics. In *Proceedings of 2006 American Control Conference*, pages 7–13. IEEE, 2006.

[10] A. Gattami and R. Murray. A frequency domain condition for stability of interconnected mimo systems. In *Proceedings of the American Control Conference*, volume 4, pages 3723 –3728 vol.4, 30 2004-july 2 2004.

[11] G.H. Golub and C.F. Van Loan. *Matrix Computations*. Johns Hopkins Univ Pr, 1996.

[12] S. Hara, T. Hayakawa, and H. Sugata. Stability analysis of linear systems with generalized frequency variables and its applications to formation control. In *Proceedings of 46th IEEE Conference on Decision and Control*, pages 1459 –1466, Dec. 2007.

[13] R. A. Horn and C. R. Johnson. *Topics in Matrix Analysis*. Cambridge University Press, 1991.

[14] F. Lashhab. Dynamic consensus networks: spectral properties, consensus, and control. *Colorado School of Mines. Arthur Lakes Library*, 2012.

[15] R. A. Horn and C. R. Johnson. *Matrix Analysis*. University Press., Cambridge, 1993.

[16] K. Kyo. Decentralized control of multi-agent systems: Theory and applications. *PhD Dissertation, Gwangju Institute of Technology, Gwangju, South Korea*, 2012.

[17] G. Lafferriere, A. Williams, J. Caughman, and JJP Veerman. Decentralized control of vehicle formations. *Systems and Control letters*, 54(9):899–910, 2005.

[18] Fadel Lashhab, Kevin Moore, Tyrone Vincent, Deyuan Meng, and Khalid Kuwairi. Robust H_∞ controller design for dynamic consensus networks. *International Journal of Control*, (just-accepted):1–20, 2017.

[19] J. Lin, AS Morse, and BDO Anderson. The multi-agent rendezvous problem. In *Proceedings of the 42nd IEEE Conference on Decision and Control*, volume 2, pages 1508–1513. IEEE, 2003.

[20] Z. Lin, B. Francis, and M. Maggiore. Necessary and sufficient graphical conditions for formation control of unicycles. *IEEE Transactions on Automatic Control*, 50(1):121–127, 2005.

[21] E. Mallada and Ao Tang. Improving damping of power networks: Power scheduling and impedance adaptation. In *Proceedings of 50th IEEE Conference on Decision and Control and European Control Conference*, pages 7729 –7734, dec. 2011.

[22] K. L. Moore, T. L. Vincent, F. Lashhab, and C. Liu. Dynamic consensus networks with application to the analysis of building thermal processes. *Proceedings of the 18th IFAC World Congress*, 18(1), 2011.

[23] R. Olfati-Saber, J.A. Fax, and R.M. Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215–233, 2007.

[24] R. Olfati-Saber and R.M. Murray. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control*, 49(9):1520–1533, 2004.

[25] R. Olfati-Saber and J.S. Shamma. Consensus filters for sensor networks and distributed sensor fusion. In *Proceedings of 44th IEEE Conference on Decision and Control and European Control Conference*, pages 6698–6703. IEEE, 2005.

[26] A. Papachristodoulou and A. Jadbabaie. Synchronization in oscillator networks: Switching topologies and non-homogeneous delays. In *Proceedings of 44th IEEE Conference on Decision and Control European Control Conference*, pages 5692–5697. IEEE, 2005.

[27] J.W. Polderman and J.C. Willems. *Introduction to Mathematical Systems Theory: a Behavioral Approach*. Springer Verlag, 1998.

[28] O. Egeland R. Lozano, B. Brogliato and B. Maschke. *Dissipative Systems Analysis and Control Theory and Application*. Springer, 2000.

[29] W. Ren, R.W. Beard, and E.M. Atkins. Information consensus in multivehicle cooperative control. *IEEE Control Systems*, 27(2):71–82, 2007.

[30] B.M. Sanandaji, T.L. Vincent, and M.B. Wakin. Exact topology identification of large-scale interconnected dynamical systems from compressive observations. In *Proceedings of American Control Conference*, pages 649–656. IEEE, 2011.

[31] H.G. Tanner, A. Jadbabaie, and G.J. Pappas. Stable flocking of mobile agents part i: Dynamic topology. In *Proceedings of the 42nd IEEE Conference on Decision and Control*, volume 2, pages 2016–2021. IEEE, 2003.

[32] H.G. Tanner, A. Jadbabaie, and G.J. Pappas. Stable flocking of mobile agents, part i: Fixed topology. In *Proceedings of the 42nd IEEE Conference on Decision and Control*, volume 2, pages 2010–2015. IEEE, 2003.

[33] J. Wang and N. Elia. Consensus over networks with dynamic channels. In *Proceedings of 2008 American Control Conference*, pages 2637–2642. IEEE, 2008.

[34] L. Xiao, S. Boyd, and S. Lall. A scheme for robust distributed sensor fusion based on average consensus. In *Proceedings of Fourth International Symposium on Information Processing in Sensor Networks*, pages 63–70. IEEE, 2005.

APPENDIX

The proof of Theorem 4.7 depends on a characterization of solutions for systems defined using a behavioral approach.

Lemma A.1 (Gershgorin cricle theorem): (see. e.g. [11]) Let $A \in \mathbb{C}^{n\times n}$ with elements a_{ij}. Let $R_i = \sum_{j\neq i} |a_{ij}|$ and define $D(a_{ii}, R_i)$ to be the closed disk centered at a_{ii} with radius R_i. Then every eigenvalue of A lies within at least one of the disks $D(a_{ii}, R_i)$.

Lemma A.2 ([27]): Given $m \times m$ polynomial matrix $P(s)$, let λ_i, $i = 1, \cdots, N$ be the distinct roots of $\det P(s)$ with multiplicity n_i. Then the solutions to

$$P\left(\frac{d}{dt}\right) x(t) = 0$$

are of the form

$$x(t) = \sum_{i=1}^{N} \sum_{j=0}^{n_i-1} \alpha_{ij} t^j e^{\lambda_i t}$$

where α_{ij} are constants.

Lemma A.3: Let the dynamic Laplacian matrix $L(j\omega) \in \mathbb{C}^{n\times n}$ and a Dynamic Grounded Laplacian $L_0(j\omega) \in \mathbb{C}^{n_0\times n_0}$ be given, with $n_0 = n - n_g < n$. For any choice of a boundary set of grounded nodes **V0**, for which $L_0(j\omega)$ is a Laplacian that corresponds to a connected graph, a lower bound for the eigenvalues of the Hermitian part of the Dynamic Grounded Laplacian matrix $\lambda_i(H(L_0(j\omega)))$ is given by

$$\lambda_i(H(L_0(j\omega))) \geq \frac{1}{\text{trace}((\text{Re}[L_0(j\omega)])^{-1})}, \qquad (69)$$

for all $\omega, i \in 1, 2, ..., n$.

Proof:

It is well known that the sum of the eigenvalues of any matrix is equal to its trace,

$$\sum_{i=1}^{n} \lambda_i(H(L_0(j\omega))^{-1}) = \text{trace}(H(L_0(j\omega))^{-1}). \tag{70}$$

Since $H(L_0(j\omega))^{-1}$ is a real and positive definite matrix (see Lemma 5.1, and Definition 4.3), then

$$\lambda_i(H(L_0(j\omega))^{-1}) > 0, \forall\omega, i = 1, 2, \ldots, n.$$

Thus,

$$\lambda_i(H(L_0(j\omega))^{-1}) \leq \text{trace}(H(L_0(j\omega))^{-1}),$$

$$\lambda_i(H(L_0(j\omega))^{-1}) \leq \text{trace}(\text{Re}[L_0(j\omega)]^{-1}). \tag{71}$$

Since the eigenvalues of $H(L_0(j\omega))$ and $[H(L_0(j\omega))]^{-1}$ are reciprocals of each other, then

$$\lambda_i([H(L_0(j\omega))]^{-1}) = \frac{1}{\lambda_i(H(L_0(j\omega)))}. \tag{72}$$

Combining (71) and (72), we get

$$\lambda_i(H(L_0(j\omega))) \geq \frac{1}{\text{trace}((\text{Re}[L_0(j\omega)])^{-1})}, \tag{73}$$

for all $\omega, i \in 1, 2, ..., n$. ■

Theorem A.4: Interlacing Eigenvalue Theorem [15]: Let A be a Hermitian matrix of order n, and let B be a principal sub matrix of A of order $n-1$. if $\lambda_{min} = \lambda_n \leq \lambda_{n-1} \leq ... \leq \lambda_2 \leq \lambda_1 = \lambda_{max}$ lists the eigenvalues of A and $\mu_n \leq \mu_{n-1} \leq ... \leq \mu_3 \leq \mu_2$ the eigenvalues of B. then

$$\lambda_n \leq \mu_n \leq \lambda_{n-1} \leq \mu_{n-1} \leq ... \leq \lambda_2 \leq \mu_2 \leq \lambda_1. \tag{74}$$

Theorem A.5: Mirsky Theorem [13]: Let $A \in \mathbb{C}^{n\times n}$ be given and set $H(A) = \frac{1}{2}(A + A^H)$. Let $\lambda_i(A)$ and $\lambda_i(H(A))$ denote the eigenvalues of A and $H(A)$, respectively, ordered so that $\text{Re}[\lambda_1(A)] \leq \text{Re}[\lambda_2(A)] \leq ... \leq \text{Re}[\lambda_n(A)]$ and $\lambda_1(H(A)) \leq \lambda_2(H(A)) \leq ... \leq \lambda_n(H(A))$. Then

$$\sum_{i=n-k}^{n} \text{Re}[\lambda_i(A)] \leq \sum_{i=n-k}^{n} \lambda_i(H(A)), \tag{75}$$

$k = 0, 1, ..., n-1$, with equality for $k = n-1$.

Lemma A.6: Let the dynamic Laplacian $L(j\omega) \in \mathbb{C}^{n\times n}$ be given. The Hermitian matrix associated with $L(j\omega)$ is $H(L(j\omega)) = \frac{1}{2}(L(j\omega) + L(j\omega)^H) = \text{Re}[L(j\omega)]$. Let $\lambda_i(L(j\omega))$ and $\lambda_i(H(L(j\omega)))$ denote the eigenvalues of $L(j\omega)$ and $H(L(j\omega))$, respectively, ordered so that $0 = \text{Re}[\lambda_1(L(j\omega))] \leq \text{Re}[\lambda_2(L(j\omega))] \leq ... \leq \text{Re}[\lambda_n(L(j\omega))]$ and $0 = \lambda_1(H(L(j\omega))) \leq \lambda_2(H(L(j\omega)))... \leq \lambda_n(H(L(j\omega)))$. Then the relationship between the real part of the smallest and largest nonzero eigenvalues of $L(j\omega)$ and $H(L(j\omega))$ can be given by

$$\text{Re}[\lambda_2(L(j\omega))] \geq \lambda_2(H(L(j\omega))), \tag{76}$$

and

$$\text{Re}[\lambda_n(L(j\omega))] \leq \lambda_n(H(L(j\omega))). \tag{77}$$

Proof:

Theorem A.5 can be applied for the matrices $L(j\omega)$ and $H(L(j\omega))$ by substituting $k = n-3$ in (75). Thus, we have

$$\sum_{i=3}^{n} \text{Re}[\lambda_i(L(j\omega))] \leq \sum_{i=3}^{n} \lambda_i(H(L(j\omega))), \tag{78}$$

with equality for $k = n-1$

$$\sum_{i=1}^{n} \text{Re}[\lambda_i(L(j\omega))] = \sum_{i=1}^{n} \lambda_i(H(L(j\omega))). \tag{79}$$

For a connected graph, $\text{Re}[\lambda_1(L(j\omega))] = \lambda_1(H(L(j\omega))) = 0$, thus we can rewrite (79) as

$$0 + \text{Re}[\lambda_2(L(j\omega))] + \sum_{i=3}^{n} \text{Re}[\lambda_i(L(j\omega))]$$

$$= 0 + \lambda_2(H(L(j\omega))) + \sum_{i=3}^{n} \lambda_i(H(L(j\omega))).$$

From the above quality and (78), we can conclude

$$\text{Re}[\lambda_2(L(j\omega))] \geq \lambda_2(H(L(j\omega))). \tag{80}$$

Now, substitute $k = 0$ in (75), we get

$$\text{Re}[\lambda_n(L(j\omega))] \leq \lambda_n(H(L(j\omega))). \tag{81}$$

■

Copyright © AJSE Publishers