A Dynamic Graph-Based Systems Framework for Modeling, and Control of Cyber-Physical Systems Typified by Buildings

Fadel Lashhab*, Musbah Abdulgader‡, and Omar A. Zargelin¶
* *Department of Electrical Engineering and Computer Science, Howard University, Washington DC, USA.*
‡ *Department of Engineering Technologies, Bowling Green State University, Bowling Green, OH, USA*
¶ *Department of Electrical and Electronic Engineering, Al Zintan University*, Al Zintan, Libya.
Email: *fadel.lashhab@howard.edu, ‡mabdulg@bgsu.edu, and ¶zargelin@uoz.edu.ly

***Abstract*—In this paper, we present a framework for modeling certain classes of cyber-physical systems using graph-theoretic thinking augmented by ideas from the field of behavioral systems theory. The cyber-physical systems we consider are typified by buildings. We show that the thermal processes associated with a building can be represented as a graph in which (1) the node variables (temperature and heat flows) are governed by a dynamic system and (2) interconnections between these nodes (walls, doors, windows) are also described by a dynamic system. In general, we call a collection of such nodes and interconnections a dynamic graph (dynamic consensus network). Motivated by building thermal example, 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 the dynamic consensus networks. Then we show how a behavioral systems approach can be used to develop kernel relationships between all system variables in dynamic graphs as typified by building thermal models. We discuss how such relationships can be used to analyze these systems' properties, focusing on controllability. The ideas developed for dynamic graph theory lead to developing a controllability analysis methodology for dynamic consensus networks in conjunction with the behavioral approach. We then developed the controllability conditions for the general dynamic networks, such as identical LTI nodes with dynamic edges or even in the more general case with heterogeneous nodes.**

***Keywords*—Dynamic graphs, dynamic consensus networks, dynamic Laplacian, Controllability analysis; Behavioral approach.**

I. Introduction

In this paper, we present a vision for the analysis and design of a class of cyber-physical systems using graph-theoretic ideas and introducing the perspective of behavioral systems theory. We are motivated by the energy-efficient control of buildings.

Our fundamental view of a building is of overlapping, interacting networks, as shown in Fig. 1. In this diagram, we depict the dominant phenomenon that contributes to the energy use of a building as networks (or graphs). The networks are made up of nodes that each represent a distinct subsystem. For example, in the thermal or human networks, the nodes may represent rooms, while in the control network, a node is a sensor, actuator, or computational unit. The links between nodes indicate variable sharing, such as the flow of people in the human network between rooms through hallways and doors or the flow of heat between rooms in the thermal network through walls and doors. Smaller circles in Fig. 1 indicate links between networks. Note that typical graph-based networks assume links that are in some way constant, but as we will see, in some networks, such as the building thermal network, links between nodes may be dynamic.

Control of distributed systems, such as shown in Fig. 1 is a currently active area of research within the field of control systems. By a distributed system, we mean one with many inputs and outputs, possibly spatially-distributed dynamics, and a decentralized decision and control architecture, with restrictions on communication between computational nodes. The current state of the art has focused primarily on homogeneous systems. However, a building may be viewed as a composite system where a physical process (the structure itself) has been augmented with a hardware infrastructure (sensors and actuators) and a cyber-infrastructure (communication and decision nodes). Such overlaid heterogeneous systems with constrained connectivity and interaction between the different layers present challenges and system optimization and control opportunities. What is needed are ways to reason about discrete, multi-attribute heterogeneous entities (such as cyber-systems) and continuous, heterogeneous processes (such as physical phenomena) operating on a hierarchy of layered graphs related to each other through a set of mappings or transformations.

In this paper, we consider methods for studying distributed systems that are heterogeneous and possibly spatially-varying. Though a building can be seen as a set of interconnected networks, we consider only the thermal network. We begin by showing how a building's thermal processes can be modeled as a graph whose node variables are temperature and heat flows and whose interconnections are walls, doors, windows, etc. In our graphical representation of a building, both the nodes and

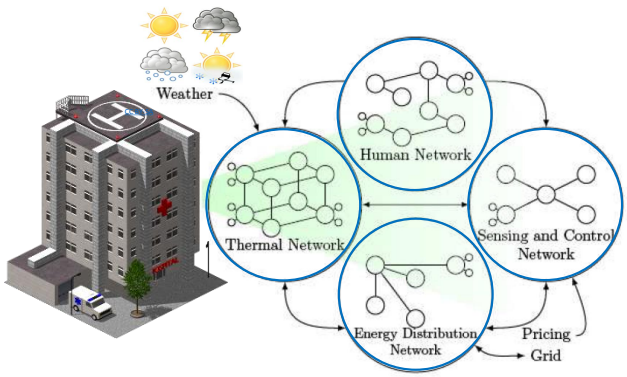

Fig. 1. A building as a collection of interacting networks.

the interconnections can be (heterogeneous) dynamic systems. We call this a *dynamic graph* (or network). For such systems, we show that the relationships between the node variables reduce to the traditional graph Laplacian in the steady-state, so that consensus variable convergence can be obtained by discussing the steady-state properties of the system. We then show how a behavioral systems approach can develop kernel relationships between all system variables in dynamic graphs typified by building thermal models. Using these kernel relationships, we consider the controllability analysis of such systems.

The idea of consensus in networking has received significant attention due to its wide array of applications in robotics, transportation, sensor networking, communication networking, biology, and physics. This paper aims to study a generalization of consensus problems whereby the weights of network edges are no longer static gains. Instead, they are dynamic systems, leading to the notion of *dynamic consensus networks*.

The network topology is static for the consensus networks, 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 [1]. Thus, static consensus problems can be written in the time domain for each node $i = 1, 2, ..., n$ as

$$\dot{x}_i = \sum_{j \in \mathcal{N}_i} \lambda_{ij}(x_j(t) - x_i(t)). \tag{1}$$

The continuous time linear consensus protocol (1) can be written in matrix form as:

$$\dot{x}(t) = -Lx(t), \tag{2}$$

where $x(t) = [x_1(t), x_2(t), ..., x_n(t)]^T$ and L, the graph's Laplacian matrix $L = [l_{ij}]$, is defined by

$$l_{ij} = \begin{cases} \sum_{j \in \mathcal{N}_i} \lambda_{ij} & i = j \\ -\lambda_{ij} & i \neq j \text{ and } (i,j) \in \mathcal{E} \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

For the multi-agent consensus problem, suppose that N agents evolve their individual beliefs $x_i \in \mathbb{R}^1$ about a so-called global consensus variable x using communications with their nearest neighbors according to the consensus protocol (1). A key result is that the solution of $\dot{x}(t) = -Lx(t)$ gives $x_i \to x^*$ if the static graph is connected [1]. This specific fact has been the basis of much of the literature related to consensus problems.

There have been many engineering scientists in the past years involved in the controllability of dynamic consensus networks. The focus was on controlling dynamic consensus networks under the leader-follower approach, where some nodes are considered leaders, and other nodes are followers. This approach aims to transfer followers' trajectories from an initial position to the desired position (set-point) by adequately selecting the leaders' trajectory. Many authors [24], [25], [26], [27], and [28] have considered this framework by using some algebraic methods and the eigenvalues and eigenvectors of the dynamic Laplacian matrix. Other researchers also investigated the controllability using graphic tools such as the graph's equitable partition [27] and symmetry properties [26]. These graphical tools are built based on the graph's configuration and topology associated with the consensus network. The controllability investigation using the minimum energy for static consensus networks using the first-order system formulated and proposed in [29], and [30]. This paper will investigate the controllability for dynamic consensus networks with edges (links) of rational dynamical systems.

Several researchers have already studied controllability analysis for consensus networks with static topology. Most of these studies have investigated the effect of the static topology on the controllability of consensus networks. The authors [11] introduced a graph-theoretic characterization of static networks' structural controllability with a single leader. They showed that a static network with a switching topology is structurally-controllable if the union graph of the underlying static topologies is connected. In [12], the controllability investigated using the graph's size and connectivity. Controllability for leader-based, multi-agent systems analyzed in [13], and [31] based on connectivity and the null space of the leader and followers' incidence matrices. Controllability using the graph symmetry, and equitable partition properties addressed in [14]. The paper [23] formulated an equivalent data-driven Hautus-type test for a general input/output system that assumes no knowledge of the system's state. The authors' work proposed in this paper also provided an algorithm for data-driven verification of controllability of the system. They used the singular value decomposition of the Hankel matrix. A multi-vehicle system's consensus problem was proposed and analyzed by [32] with a time-varying reference state. Under the condition, only a portion of the vehicles can access the reference state in this problem. Those vehicles might not have the ability to share the information with the other vehicles in the team. Although their paper focused on developing an algorithm for investigating the consensus conditions for a directed fixed information-exchange topology, so it is useful to extend this algorithm to directed switching information-exchange topologies. In our article, the topology (edges) that describes the interconnections between nodes is considered time-varying rational transfer functions. Investigating the consensus conditions and the controllability for a multi-vehicle system might be one of the motivating application of this work.

The consensus protocol such as in (1) and its variants have been studied extensively in the literature. They have been applied in many areas, notably for formation motion control when the agents are mobile. However, the consensus paradigm is restrictive in several ways. Notice that we have interpreted the consensus problem as having integrating nodes and static weights. One might ask: what if the weights were also transfer functions? What if the nodes are more than integrators? In the next section, we have considered this question, motivated by modeling heat transfer in buildings [2]–[5]. By the notation "dynamic systems," we mean that linear ordinary differential equations (LODEs) are described as relationships between the system variables. We call such networks *dynamic consensus networks* because all the node variables converge to a common value called a consensus value under some conditions.

The paper is organized as follows: In Section II, we present a general framework for a dynamic consensus network. We present a detailed study of modeling thermal processes in buildings as directed, dynamic graphs, beginning with a simple two-room model and transitioning to a model with multiple interconnected rooms. Section III presents 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. In Section IV, we use the established aspects and properties of the defined dynamic graph theory in conjunction with the behavioral approach to developing a controllability-analysis methodology for dynamic networks. Because of scalability in such dynamic networks, the controllability conditions based on node and network topology for the overall dynamic networks were developed in Section V.

II. Modeling a Thermal Process in a Building as a Directed Dynamic Graph

This Section first presents examples showing how a dynamic graph can arise in applications and then give a general framework for a dynamic consensus network. We present a detailed study of modeling thermal processes in buildings as directed, dynamic graphs, beginning with a simple two-room model and transitioning to a model with multiple interconnected rooms. Motivated by this example, we then 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.

Historically, there has always been a recognized need to model the energy processes within buildings. Typical examples of this modeling application are sizing HVAC equipment, determining energy usage performance, and optimizing energy management in a building through persistent control. Current state-of-the-art methods include modeling packages, such as Energy Plus [6], that allows users to specify a building's geometry, equipment, orientation, materials, and usage patterns, which are then simulated using first principles models and simulated weather data. Though undoubtedly useful for design, these computationally-complex systems may suffer from certain limitations once a building has been constructed due to significant deviations in construction, occupant use, and other specifications that cause the actual building's behavior to be quite different from the model.

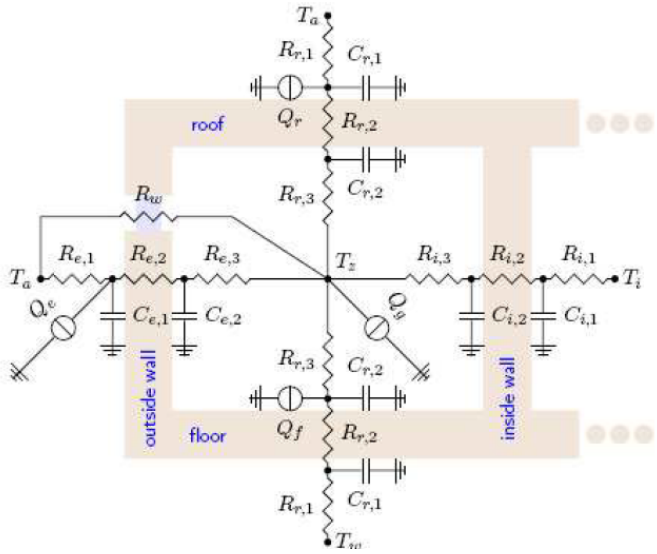

Fig. 2. Thermal model of a room.

At the opposite extreme, so-called black box models have been developed from observational data. Though these models can be utilized to predict future values of particular variables, they do not incorporate any structural information about the system when gathering data, resulting in the need for large amounts of data to train and suffering from the difficulty of extracting relevant information about internal physical parameters that may be of interest.

Semi-physical models resulting in an intermediate level of modeling are known as gray-box modeling. Simple modeling elements containing parameters identified using observational data are chosen and connected based upon physical insight to represent the system's actual configuration. This is commonly the modeling technique used for thermal networks, which have been used to study load-shifting and peak-reducing control in buildings [7], [8]. A typical thermal network model for a single room is shown in Fig. 2, which was adapted from [8]. These networks of (analogous) thermal resistors and capacitors model different building elements. To date, this has typically been performed at a very coarse level, sometimes by combining multiple rooms into one practical room per zone. In [8], a gray-box model for an experimental building was created by utilizing measurements of weather, room temperature, and room air supply and flow. This model was used to predict the effects of a demand-limiting control strategy that was later validated experimentally.

This section uses a single-room model as shown in Fig. 2 from [8] as the basis for a node and its interconnections to

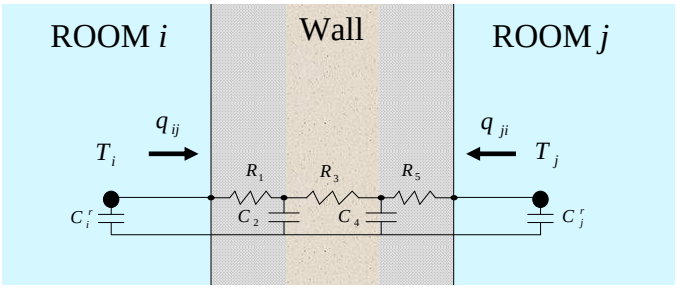

Fig. 3. Two rooms connected by a wall using the 3R2C model.

other nodes to build up a dynamic graph representation of a building's thermal processes. First, we consider two rooms connected by a wall. We then illustrate how several such nodes may be interconnected, using the example of a hypothetical four-room building, with analysis provided of the resulting model that motivates the generalization in the next section.

Before proceeding, we note that the initial interest in modeling thermal processes in a building comes from viewing a building as a group of overlapping, interacting networks. In the thermal network, the nodes may represent rooms, while in the control network, a node is a sensor, actuator, or computational unit. The links between nodes indicate variable information sharing, such as the heat flow between rooms through walls and doors in the thermal network. The typical, graph-based networks assume links that are in some way constant; some networks, such as a building's thermal network, may have dynamic links between nodes, as we will see in the next section.

A. Two Rooms Connected by a Wall

Fig. 3 depicts what is called a 3R2C model in the literature [9]. We identify a room i as a node with node variable T_i, the lumped room temperature, Q_i^{in}, the input heat flow (a manipulated variable, not shown), and q_{ij}, the heat flow out of the room through walls or doors or windows (of course with this convention, if $q_{ij} < 0$ then this is heat flow into the room). The parameter C_i^r is the thermal capacity (mass) of the room i.

The interconnection between the two rooms is a wall, which is represented analogously by an electrical circuit with three resistors and two capacitors, a simplification of the model in Fig. 3. The capacitors C_2 and C_4 can be thought of as the heat storage capacity of the wall's materials, which could be different on each side of the wall. A complete model adds a capacitor to represent the insulation properties in addition to that of the wall's board materials. For the resistors, R_3 represents the heat dissipation inside the wall, while R_1 and R_5 represent the heat dissipation from each room to the inside of the wall.

As shown in [9], the heat flows in Fig. 3 can be written as:

$$\begin{bmatrix} q_{ij} \\ q_{ji} \end{bmatrix} = \frac{1}{B_{ij}(s)} \begin{bmatrix} A_{ij}(s) & -D_{ij}(s) \\ -D_{ij}(s) & A_{ji}(s) \end{bmatrix} \begin{bmatrix} T_i \\ T_j \end{bmatrix} \tag{4}$$

where

$$\begin{aligned} A_{ij}(s) &= 1 + a_1^{ij} s + a_2^{ij} s^2 \\ A_{ji}(s) &= 1 + a_1^{ji} s + a_2^{ji} s^2 \\ B_{ij}(s) &= b_0^{ij} + b_1^{ij} s + b_2^{ij} s^2 \\ D_{ij}(s) &= 1 + d_1^{ij} s + d_2^{ij} s^2 \end{aligned}$$

and

$$\begin{aligned} a_1^{ij} &= C_4R_5 + C_2R_3 + C_2R_5 \\ a_2^{ij} &= C_4C_2R_5R_3 \\ a_1^{ji} &= C_4R_1 + C_4R_3 + C_2R_1 \\ a_2^{ji} &= C_4C_2R_3R_1 \\ b_0^{ij} &= R_5 + R_3 + R_1 \\ b_1^{ij} &= C_4R_5R_1 + C_2R_3R_1 + C_2R_5R_1 + C_4R_5R_3 \\ b_2^{ij} &= C_4C_2R_5R_3R_1 \\ d_1^{ij} &= d_2^{ij} = 0 \end{aligned}$$

Here, s is the independent variable of the Laplace transform, which can be interpreted as $s \doteq \frac{d}{dt}(\cdot)$. As noted in [9], we can interpret $G_x(s) = A_{ij}(s)/B_{ij}(s)$ as the external conduction of the wall, $G_y(s) = D_{ij}(s)/B_{ij}(s)$ as the cross-conduction of the wall, and $G_z(s) = A_{ji}(s)/B_{ij}(s)$ as the internal conduction of the wall.

From (3), the nodal equation can be written as:

$$C_i^r \frac{dT_i}{dt} = Q_i^{in} - q_{ij} \tag{5}$$

Combining (4) and (5) gives:

$$\begin{bmatrix} C_i^r s & 0 \\ 0 & C_j^r s \end{bmatrix} \begin{bmatrix} T_i \\ T_j \end{bmatrix} = \begin{bmatrix} Q_1^{in} \\ Q_2^{in} \end{bmatrix} - \frac{1}{B_{ij}(s)} \begin{bmatrix} A_{ij}(s) & -D_{ij}(s) \\ -D_{ij}(s) & A_{ji}(s) \end{bmatrix} \begin{bmatrix} T_i \\ T_j \end{bmatrix} \tag{6}$$

To motivate later analysis, notice that in the absence of any external heat inputs (i.e., $Q_i^{in} = 0$), we can rewrite the previous equation as:

$$\begin{bmatrix} sC_i^r T_i(s) \\ sC_i^r T_j(s) \end{bmatrix} = -\frac{1}{B_{ij}(s)} \begin{bmatrix} A_{ij}(s) & -D_{ij}(s) \\ -D_{ij}(s) & A_{ji}(s) \end{bmatrix} \begin{bmatrix} T_i \\ T_j \end{bmatrix}, \tag{7}$$

which defines the relationship between the temperatures in two rooms using the 3R2C model.

It is useful to separate (7) as

$$\begin{aligned} T_i(s) &= -\frac{1}{C_i^r s}\left[\frac{A_{ij}(s)}{B_{ij}(s)} T_i(s) - \frac{D_{ij}(s)}{B_{ij}(s)} T_j(s)\right] \\ &= -\frac{1}{s}[\lambda_{ij}^S(s) T_i(s) - \lambda_{ij}^C(s) T_j(s)], \end{aligned} \tag{8}$$

where $\lambda_{ij}^S(s) = \frac{A_{ij}(s)}{B_{ij}(s)}$ is a self-correction weight and $\lambda_{ij}^C(s) = \frac{D_{ij}(s)}{B_{ij}(s)}$ is a cross-correction weight. Note that we assume C_i^r in (8) is equal to the unity for simplicity. Comparing this to static-weights consensus networks, we see that this appears similar to the equation of a two-node consensus network, in which the nodes are integrators. The difference is that there is a separate weighting on the terms $T_i(s)$ and $T_j(s)$, and these weights are dynamic.

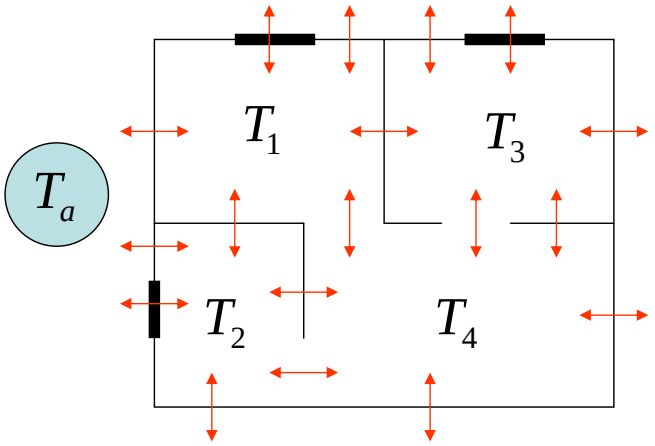

Fig. 4. A hypothetical four-room example.

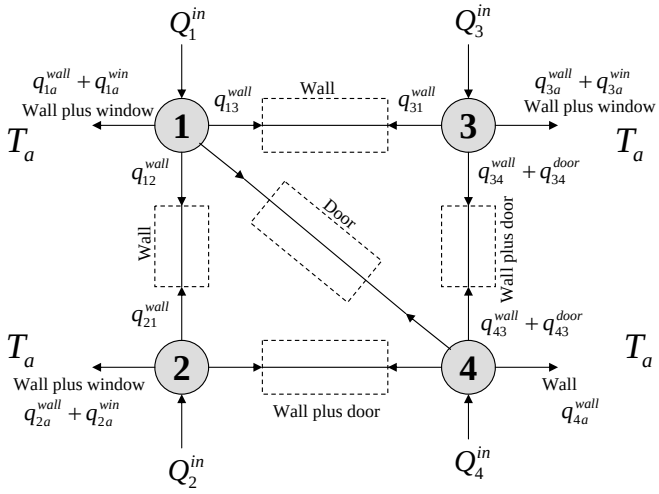

Fig. 5. Heat flow network corresponding to the four-room example.

B. Several Interconnected Rooms

This subsection uses the previous subsection's expressions to develop a building model with several interconnected rooms with different possible pathways between each room and the outside environment. Ideas are developed for the specific hypothetical four-room building shown in Fig. 4 in which each room has several neighbors with which it is interconnected. One such neighbor is always the external environment whose variable is denoted T_a with $'a'$ referring to the ambient. Pathways include walls, doors, and windows. The corresponding graph for this example is shown in Fig. 5. There are several heat flows that are not shown, including $q_{14} = q_{14}^{door}, q_{41} = q_{41}^{door}, q_{24} = q_{24}^{wall} + q_{24}^{door}$ and $q_{42} = q_{42}^{wall} + q_{42}^{door}$.

In developing a model for this system, we modify (5) to sum the energy losses through all pathways connected to a node, resulting in:

$$C_i^r \frac{dT_i}{dt} = Q_i^{in} - \sum_{j \in \mathcal{N}_i} \sum_{k_j \in \mathcal{P}_j} q_{ij}^{k_j} \tag{9}$$

where $\mathcal{N}_i$ is the set of neighbors to which a node i is connected and $\mathcal{P}_j$ is the set of pathways k_j associated with any neighbor j of node i. We note that $q_{ij} = \sum_{k_j \in \mathcal{P}_j} q_{ij}^{k_j}$.

For building thermal analysis, there may be several different types of interconnection elements, though they will all have the basic format

TABLE I
HYPOTHETICAL FOUR ROOM EXAMPLE

Node	Neighbors	Paths	Coefficients
1	2	1-wall	A_{12}, B_{12}
	a	2-wall	A'_{1a}, B'_{1a}
		1-window	R_{w1a}
	3	1-wall	A_{13}, B_{13}
	4	1-door	$R_{d14} = B_{14}$
2	1	1-wall	D_{12}, B_{12}
	a	2-wall	A'_{2a}, B'_{2a}
		1-window	$R_{w2a} = B_{2a}$
	4	2-wall	A'_{24}, B'_{24}
		1-door	$R_{d24} = B_{24}$
3	1	1-wall	D_{13}, B_{13}
	a	2-wall	A'_{3a}, B'_{3a}
		1-window	$R_{w3a} = B_{3a}$
	4	2-wall	A'_{34}, B'_{34}
		1-door	$R_{d34} = B_{34}$
4	1	1-door	$R_{d41} = B_{41}$
	a	1-wall	A'_{4a}, B'_{4a}
	2	2-wall	D'_{24}, B'_{24}
		1-door	$R_{d42} = B_{42}$
	3	2-wall	D'_{34}, B'_{34}
		1-door	$R_{d43} = B_{43}$

of (4). Because there is negligible energy storage in doorways and windows, when these are the sole interconnection elements between rooms, we use a single R model, so that (4) is expressed with $b_0^{ij} = R$ with all other variables being set to zero. To make the notation a bit more uniform, for the common case when a door or window is in parallel with a wall, the interconnection transfer function matrix is the sum of the single R model and the 3R2C model. That is, the model would be given by A_{ij},A_{ji}, B_{ij}, and D_{ij} that satisfies:

$$\begin{bmatrix} \frac{A_{ij}(s)}{B_{ij}(s)} & -\frac{D_{ij}(s)}{B_{ij}(s)} \\ -\frac{D_{ij}(s)}{B_{ij}(s)} & \frac{A_{ji}(s)}{B_{ij}(s)} \end{bmatrix} = \begin{bmatrix} \frac{A'_{ij}(s)}{B'_{ij}(s)} & -\frac{1}{B'_{ij}(s)} \\ -\frac{1}{B'_{ij}(s)} & \frac{A'_{ji}(s)}{B'_{ij}(s)} \end{bmatrix} + \begin{bmatrix} \frac{1}{R} & -\frac{1}{R} \\ -\frac{1}{R} & \frac{1}{R} \end{bmatrix}, \tag{10}$$

where the primed variables represent the 3R2C model, and the unprimed variables represent the resulting parallel connection. In the expressions below, we assume that this computation has been done and the resulting unprimed coefficients can be easily calculated and are thus omitted here. Note that in the case of a door or window that is parallel to a wall, the coefficients d_1^{ij} and d_2^{ij} associated with the unprimed variables are non-zero.

Table I summarizes the neighbors for each node and the pathways between each node and each of its neighbors for this example. The table also identifies the coefficients used in the transfer matrix describing the interconnection between each pair of neighbors where the various polynomials A_{ij}, B_{ij}, D_{ij} (when there is no parallel door or wall) or $A'_{ij}, B'_{ij}, D'_{ij}$ (before being combined in parallel with any doors or windows) are defined as above in (4).

Combining (4) and (9) for the configuration shown in Fig. 5 with the parameters shown in Table I and defining the vectors

$$T(s) = [T_1(s) \quad T_2(s) \quad T_3(s) \quad T_4(s)]^T,$$
$$Q^{in}(s) = [Q_1^{in}(s) \quad Q_2^{in}(s) \quad Q_3^{in}(s) \quad Q_4^{in}(s)]^T,$$

we can easily show that:

$$sT(s) = Q^{in}(s) - L(s)T(s), \tag{11}$$

where the matrix $L(s) = [L_{ij}(s)]$ is given as:

$$L_{ij}(s) = \begin{cases} \sum_{j \in \mathcal{N}_i} \lambda_{ij}^S(s) & i = j \\ -\lambda_{ij}^C(s) & i \neq j \text{ and } (i,j) \in \mathcal{E} \\ 0 & \text{otherwise,} \end{cases} \tag{12}$$

or $L(s)$ is given as (13). We will refer to $L(s)$ defined in this way as a *dynamic Laplacian matrix*. For the graph topology shown in Fig. 5, the dynamic Laplacian matrix has the form shown in (13). In the previous work [2], We have shown that when the weight matrices $\lambda_{ij}(s)$ satisfy certain assumptions, $-L(s)$ can be viewed as a *dynamic interconnection matrix*, allowing the demonstration of consensus.

Notice that we can redraw Fig. 5 as shown in Fig. 6, where

$$\lambda_{ij}(s) = \begin{bmatrix} \lambda_{ij}^S(s) & -\lambda_{ij}^C(s) \end{bmatrix} \tag{14}$$

$$= \left[\left(\frac{A_{ij}}{B_{ij}}\right)_w + \frac{1}{R_{ij}^d} \quad -\left(\frac{1}{B_{ij}}\right)_w - \frac{1}{R_{ij}^d} \right], \tag{15}$$

The graph shown in Fig. 6 will be referred to as a *dynamic graph* or a *dynamic consensus network*. Then applying the definition of the dynamic Laplacian (12) for the dynamic graph Fig. 6 we get

$$L(s) = \begin{bmatrix} \sum_{j=2,3,4} \lambda_{1j}^S(s) & -\lambda_{12}^C(s) & -\lambda_{13}^C(s) & -\lambda_{14}^C(s) \\ -\lambda_{21}^C(s) & \sum_{j=1,4} \lambda_{2j}^S(s) & 0 & -\lambda_{24}^C(s) \\ -\lambda_{31}^C(s) & 0 & \sum_{j=1,4} \lambda_{3j}^S(s) & -\lambda_{34}^C(s) \\ -\lambda_{41}^C(s) & -\lambda_{42}^C(s) & -\lambda_{43}^C(s) & \sum_{j=1,2,3} \lambda_{4j}^S(s) \end{bmatrix}, \tag{16}$$

which reduces to (13) if we insert the full expressions for $\lambda_{ij}^C(s)$ and $\lambda_{ij}^S(s)$ defined in (14). This leads us to consider the idea of dynamic consensus networks.

Figure II-B shows a simple simulation of (9) for the case when $Q_i^{in} = 0$ and the scalar $T_a = 80$ for a nominal set of parameters available from the authors upon request (omitted here in the interest of space). As intuitively expected, all temperatures converge to the ambient temperature, as the external environment has infinite capacity and there is no energy input or removal. This can also be seen by examining (9) at steady-state with $Q_{in} = 0$. Further noting that L(0) has the form of a classic graph Laplacian matrix, with row sum equal to zero (and in this case column sum equal zero as well), we can argue that $T^{ss} = T_a$ is a unique solution.

We also [10] consider another example that motivated a generalization of the static consensus problem (1), modeling the load frequency control (LFC) network of an electrical power grid as a dynamic consensus network. We consider the following network:

$$Y_i(s) = \frac{1}{s} \sum_{j \in \mathcal{N}_i} G_i(s) a_{ij} (Y_j(s) - Y_i(s)), \tag{17}$$

$i = 1, \ldots, N$, which can be viewed as a single-integrator consensus network with dynamic interconnection coefficients $G_i(s)a_{ij}$. In the grid's LFC network, each system's output is the phase of its voltage, which is the integration of the angular velocity. The interconnection is power exchanges among the individual systems through transmission lines dependent on phase differences.

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,

$$x_i(s) = -\frac{1}{s} \sum_{j \in \mathcal{N}_i} \left[\lambda_{ij}^S(s) x_i(s) - \lambda_{ij}^C(s) x_j(s) \right] \tag{18}$$

- **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,

$$x_i(s) = -\frac{1}{s} \sum_{j \in \mathcal{N}_i} \left[\lambda_{ij}(s)(x_i(s) - x_j(s)) \right] \tag{19}$$

- **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,

$$x_i(s) = -p(s) \sum_{j \in \mathcal{N}_i} \left[\lambda_{ij}(s)(x_i(s) - x_j(s)) \right] \tag{20}$$

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

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

$$x_i(s) = -p_i(s) \sum_{j \in \mathcal{N}_i} \left[\lambda_{ij}(s)(x_i(s) - x_j(s)) \right] \tag{21}$$

Assumptions: For the dynamic networks (18 - 21), we make the following assumptions:

1) The node and edge processing in the proposed dynamic networks (18 - 21) 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 (19), we assume the edges' dynamics are strictly positive real (SPR) transfer functions.
3) The topology of a network can be directed or undirected. The first dynamic network (18) uses a directed topology, whereas the second dynamic network (19) 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 Section. The difference between these two cases is illustrated in (18, 19). These flow models are rooted in the types of dynamic networks to be modeled per the motivation for each network. The first dynamic network (18) is based upon modeling buildings' thermal processes as directed dynamic graphs. In contrast, the second dynamic network (19) is based upon the motivation of modeling micro-grids of power systems as undirected dynamic graphs.
5) The nodes' dynamics can be integrators (18, 19) or more general dynamics (20, 21).
6) The nodes' dynamics and the edges can be identical (20) or heterogeneous (21).
7) These models are often autonomous, meaning no input flows into the dynamic consensus networks. However, we add inputs and disturbances to the proposed dynamic consensus networks' general forms in some problems.

III. Dynamic Graphs Definitions

The previous work [2], [10] showed how two different phenomena could be modeled in a graph whose edges are transfer functions (i.e., dynamic systems). In this section, from the models' motivation 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 the static graph.

Consider the example of a directed, dynamic graph shown in Fig. 8. Such graphs can be described as a set of nodes (or vertices) $\mathcal{N} =$

$$L(s) = \begin{bmatrix} \frac{1}{R_{14}^d} + \sum_{j=2,3} \frac{A_{1j}}{B_{1j}} & -\frac{1}{B_{12}} & -\frac{1}{B_{13}} & -\frac{1}{R_{14}^d} \\ -\frac{1}{B_{21}} & \frac{1}{R_{24}^d} + \sum_{j=1,4} \frac{A_{2j}}{B_{2j}} & 0 & -\frac{1}{B_{24}} - \frac{1}{R_{24}^d} \\ -\frac{1}{B_{31}} & 0 & \frac{1}{R_{34}^d} + \sum_{j=1,4} \frac{A_{3j}}{B_{3j}} & -\frac{1}{B_{34}} - \frac{1}{R_{34}^d} \\ -\frac{1}{R_{41}^d} & \frac{-1}{B_{42}} - \frac{1}{R_{42}^d} & -\frac{1}{B_{43}} - \frac{1}{R_{43}^d} & \frac{1}{R_{41}^d} + \frac{1}{R_{42}^d} + \frac{1}{R_{43}^d} + \sum_{j=2,3} \frac{A_{4j}}{B_{4j}} \end{bmatrix}. \quad (13)$$

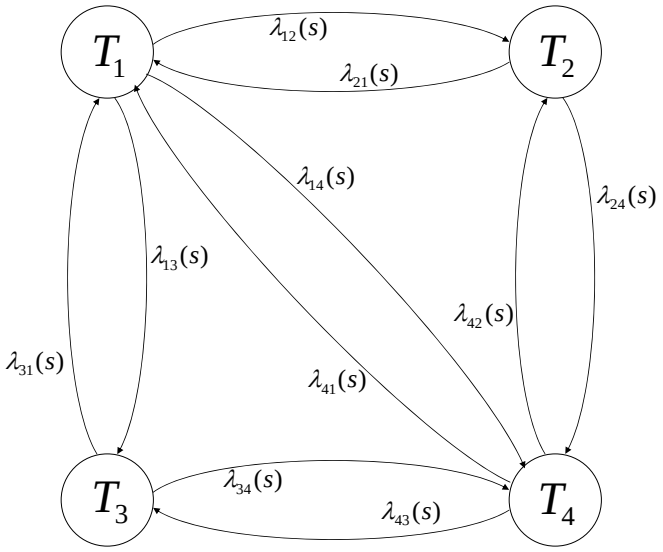

Fig. 6. A hypothetical four-room example as a dynamic consensus network.

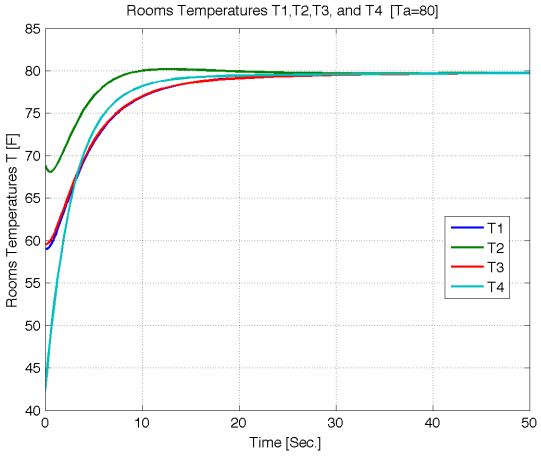

Fig. 7. Example simulation.

$\{n_i\}$ 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

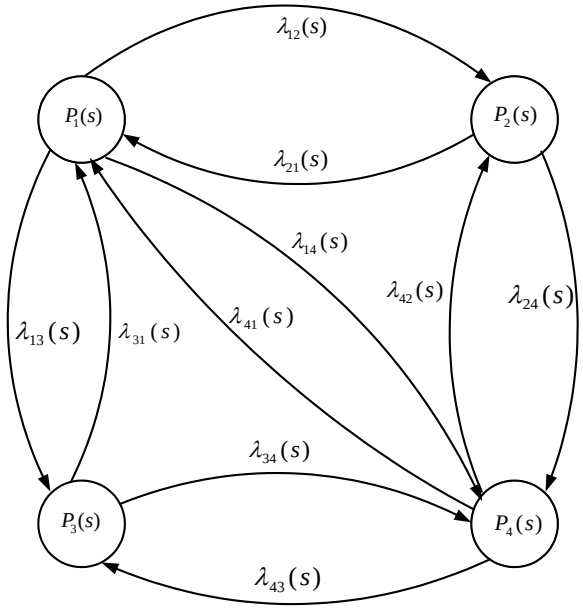

Fig. 8. Directed-Dynamic Graph.

between two nodes is a sequence of edges by which it is possible to move along the arc sequence from one of the nodes to the other. If there is at least one node with at least one path to every other node, the graph is 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 the 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 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 particular case is when we have integrator nodes with dynamic edges.

Each node $P_i(s)$ in a directed dynamic graph, such as Fig. 8, 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

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

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

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

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,

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

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

Similarly, we can define a dynamic and static outgoing incident matrices for a dynamic graph by, $B_D^{out}(s) = [b_{ij-D}^{out}(s)]$, $B_S^{out} = [b_{ij-S}^{out}]$, where,

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

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

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 $B_D(s) = B_D^{in}(s) + B_D^{out}(s)$ and $B_S = B_S^{in} + B_S^{out}$.

We can now give the dynamic equivalent of the static Laplacian matrix with the derived dynamic degree, adjacency, and incident matrices. 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

$$l_{ij}(s) = \begin{cases} \sum_{j\in\mathcal{N}_i} \lambda_{ij}(s) & i = j \\ -\lambda_{ij}(s) & i \neq j \text{ and } (i,j) \in \mathcal{E} \\ 0 & \text{otherwise} \end{cases}. \tag{22}$$

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

$$L(s) = BD(s)B^T = D(s) - A(s), \tag{23}$$

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:

$$L^{out}(s) = D^{out}(s) - A^{out}(s), \tag{24}$$

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:

$$\begin{aligned} D^{out}(s) &= B_D^{out}(s){B_S^{out}}^T, \\ A^{out}(s) &= -B_D^{out}(s){B_S^{in}}^T. \end{aligned} \tag{25}$$

By combining (24) and (25), the outgoing, dynamic Laplacian matrix $L^{out}(s)$ can be defined as

$$\begin{aligned} L^{out}(s) &= B_D^{out}(s){B_S^{out}}^T + B_D^{out}(s){B_S^{in}}^T \\ &= B_D^{out}(s)({B_S^{out}}^T + {B_S^{in}}^T) = B_D^{out}(s)B_S^T. \end{aligned} \tag{26}$$

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

To illustrate, for the example shown in Fig. 8, the associated dynamic degree, adjacency, and Laplacian matrices are given by (27). Note that all $\lambda_{ij}(s)$ in (27) are transfer functions that describe the interconnections (edges) between the nodes.

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

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

$$L^{out}(s) = \begin{bmatrix} \lambda_{12}(s)+\lambda_{13}+\lambda_{14}(s) & -\lambda_{12}(s) & -\lambda_{13}(s) & -\lambda_{14}(s) \\ -\lambda_{21}(s) & \lambda_{21}(s)+\lambda_{24}(s) & 0 & -\lambda_{24}(s) \\ -\lambda_{31}(s) & 0 & \lambda_{31}(s)+\lambda_{34}(s) & -\lambda_{34}(s) \\ -\lambda_{41}(s) & -\lambda_{42}(s) & -\lambda_{43}(s) & \lambda_{41}(s)+\lambda_{42}(s)+\lambda_{43}(s) \end{bmatrix}. \quad (27)$$

IV. Controllability Analysis for Dynamic Consensus Networks Using the Behavioral Approach

This section uses the established aspects and properties of the defined dynamic graph theory in conjunction with the behavioral approach to developing a controllability-analysis methodology for dynamic networks. Additionally, we demonstrate the dynamic interconnection topology (dynamic Laplacian matrix) in analyzing node interconnections to establish controllability.

A. Behavioral Approach

This section analyzes the dynamic interconnection topologies of the proposed dynamic networks to establish controllability using the behavioral approach to control [15]. We first describe the behavioral process and then discuss the two-room example's controllability utilizing this approach. We then give the general result.

The behavior of a dynamic system is the collection of all possible time trajectories in the system [16], [17]. In other words, the behavior is a family of trajectories, rather than a transfer function. The behavior of a dynamic system is defined as the set of solutions of a system of linear, constant-coefficient differential equations. For example, consider a system described by the following set of differential equations:

$$\begin{aligned} & w_1 + 2\frac{d}{dt}w_1 + \frac{d^2}{dt^2}w_1 - w_2 - \frac{d}{dt}w_2 = 0; \\ & -w_1 - \frac{d}{dt}w_1 + 7w_2 + 5\frac{d}{dt}w_2 + \frac{d^2}{dt^2}w_2 - 6w_3 - \\ & 4\frac{d}{dt}w_3 = 0; \end{aligned} \quad (28)$$

The dynamic system (28) can be written in matrix form as:

$$M\left(\frac{d}{dt}\right)w = 0, \quad (29)$$

where

$$M(\frac{d}{dt}) = \begin{bmatrix} \frac{d^2}{dt^2}(\cdot) + 2\frac{d}{dt}(\cdot) + 1 & -(\frac{d}{dt}(\cdot)+1) & 0 \\ -(\frac{d}{dt}(\cdot)+1) & \frac{d^2}{dt^2}(\cdot) + 5\frac{d}{dt}(\cdot) + 7 & -4\frac{d}{dt}(\cdot) - 6 \end{bmatrix},$$

and $w = [w_1, w_2, w_3]^T$. We can also describe (28) using polynomial notation as

$$\begin{bmatrix} s^2+2s+1 & -(s+1) & 0 \\ -(s+1) & s^2+5s+7 & -4s-6 \end{bmatrix}\begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} = 0, \quad (30)$$

From the above equations, we can define the behavior of the system (28) as all possible sets of solutions (w_1, w_2, and w_3) that satisfy (29) or (30). We call equation (29) or (30) a kernel representation of the dynamic system (28) [15].

We can use the behavioral system theory notation $M(\frac{d}{dt})$ to describe the relationships between the variables at a node can be represented by a kernel operator made up of differential equations. For more details, see [18]. To illustrate, given a node with the transfer function description:

$$T_i(s) = -\frac{10}{s+1}q_{i1}(s) - \frac{s+4}{s+2}q_{i2}(s) + \frac{s+5}{s+1}Q_i^{in}(s),$$

which could be rewritten as

$$T_i(s) + \frac{10}{s+1}q_{i1}(s) + \frac{s+4}{s+2}q_{i2}(s) - \frac{s+5}{s+1}Q_i^{in}(s) = 0,$$

we can describe this as $M(\frac{d}{dt})x(t) = 0$, where $x(t) = [T_i(t)\ q_{i1}(t)\ q_{i2}(t)\ Q_i^{in}(t)]^T$ and $M(\frac{d}{dt})$ is defined by (31).

With this notation, an edge can be represented as $M^{i\leftrightarrow j}(\frac{d}{dt})$, as is done in considering the two-node example shown in Fig. 9. The transfer function in the connection element is equivalently:

$$\begin{bmatrix} -(\frac{d}{dt}(\cdot)+3) & 0 & \frac{d}{dt}(\cdot)+1 & 10 \\ 0 & -(\frac{d}{dt}(\cdot)+3) & 10 & \frac{d}{dt}(\cdot)+2 \end{bmatrix}\begin{bmatrix} q_1 \\ q_2 \\ T_1 \\ T_2 \end{bmatrix} = 0. \quad (32)$$

The kernel representation for the system can then be defined by:

$$M\left(\frac{d}{dt}\right)x(t) = 0. \quad (33)$$

B. Controllability using the Behavioral Representation

Definition 4.1: [15] The system (33) is *controllable* if for any two trajectories $x_1(t)$ and $x_2(t)$ (as shown in Fig. 10) satisfying (33) there exists a time $T \geq 0$ and a third trajectory $x_3(t)$ satisfying (33) such that

$$x_3(t) = \begin{cases} x_1(t) & t \leq 0 \\ x_2(t-T) & t \geq T \end{cases}$$

In a controllable system, it is possible to utilize freely-assignable variables to switch between any legal past and future trajectories with some delay T as shown in Fig. 10 . Thus, for a Controllable system, it is possible to reach any legal future trajectory regardless of its present state or to use the transition trajectory. In [19], controllability was described using the kernel representation:

Theorem 4.1:

The system (33) is controllable if and only if the rank of $M(s)$ is the same for all $s \in \mathbb{C}$.

$$M(\frac{d}{dt}) = [(\frac{d^2}{dt^2}(\cdot) + 3\frac{d}{dt}(\cdot) + 2) \quad 10(\frac{d}{dt}(\cdot) + 2) \quad (\frac{d^2}{dt^2}(\cdot) + 5\frac{d}{dt}(\cdot) + 4) \quad -(\frac{d^2}{dt^2}(\cdot) + 7\frac{d}{dt}(\cdot) + 10)] \tag{31}$$

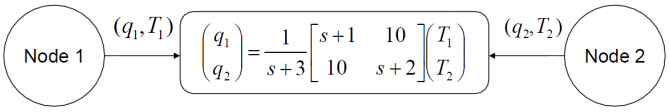

Fig. 9. Connection transfer function example.

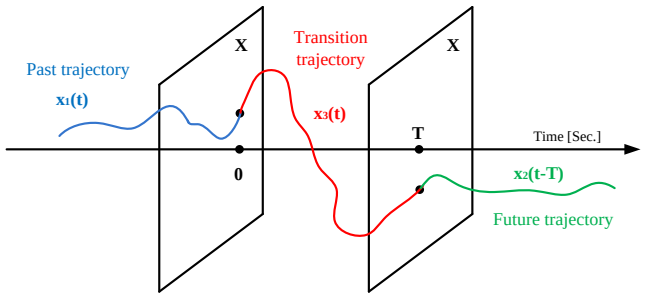

Fig. 10. Behavioral controllability.

C. Controllability of Two Rooms: An Example

As an example, we investigate the controllability of the two-room example. Our problem is first formulated by defining the behavioral representation of the model containing two rooms (For more details in the modeling of two rooms example, see Section II-A. From Fig. 3, the nodal equation can be written as:

$$\begin{bmatrix} C_i^r s & 0 \\ 0 & C_j^r s \end{bmatrix}\begin{bmatrix} T_i \\ T_j \end{bmatrix} = \begin{bmatrix} Q_1^{in} \\ Q_2^{in} \end{bmatrix} - \frac{1}{B_{ij}(s)}\begin{bmatrix} A_{ij}(s) & -D_{ij}(s) \\ -D_{ij}(s) & A_{ji}(s) \end{bmatrix}\begin{bmatrix} T_i \\ T_j \end{bmatrix} \tag{34}$$

From (34), the kernel representation of the two rooms example can be expressed by (35).

Using Theorem 4.1 and the structures of the system's matrices, the following theorem defines the controllability of the two rooms example:

Theorem 4.2: Suppose for all i, j, the roots of $B_{ij}(s)$ (in 6) are isolated. Then system (35) is uncontrollable if and only if there exists a root λ of $B_{ij}(s)$ such that

$$\begin{bmatrix} A_{ij}(\lambda) & -D_{ij}(\lambda) \\ -D_{ij}(\lambda) & A_{ji}(\lambda) \end{bmatrix} \tag{36}$$

is not full row rank.

Proof:

Based on Theorem

Case 1: λ not a root of $B_{ij}(s)$: The system is controllable because $B_{ij}(\lambda) \neq 0$, thus the rank of $M(s)$ is two.

Case 2: λ a root of $B_{ij}(s)$: In this case $B_{ij}(\lambda) = 0$, so $M(s)$ becomes

$$M(\lambda) = \begin{bmatrix} A_{ij}(\lambda) & -D_{ij}(\lambda) & 0 & 0 \\ -D_{ij}(\lambda) & A_{ji}(\lambda) & 0 & 0 \end{bmatrix}. \tag{37}$$

The last equation shows that the system is uncontrollable if and only if the non-zeros block matrix.

$$\begin{bmatrix} A_{ij}(\lambda) & -D_{ij}(\lambda) \\ -D_{ij}(\lambda) & A_{ji}(\lambda) \end{bmatrix} \tag{38}$$

is not full row rank. ■

A similar result is also possible with repeated roots where the rank condition would have to be verified for terms that share roots of the same or higher multiplicity.

V. General Controllability of Dynamic Consensus Networks

The controllability conditions obtained in the previous section cannot be applied for the general dynamic networks, such as identical LTI nodes with dynamic edges or even in the more general case with heterogeneous nodes. This is because of scalability in such dynamic networks. For example, we have a similar result for the four rooms example, but this result will be challenging to apply when considering more rooms or nodes in a dynamic network. Thus, we seek to develop controllability conditions based on node and interconnection (edge) parameters that guarantee the overall dynamic network's controllability.

The block diagram for the general dynamic networks, heterogeneous nodes with dynamic edges proposed in (21) can be depicted in Fig.reffig:DynamicHetrogenuousNetwork. To propose our task, our problem is first formulated by considering a dynamic consensus network consisting of N heterogeneous LTI nodes $P_i(s)$ for all $i = 1, \ldots, N$ modeled by the following state-space equations:

$$\begin{aligned} \dot{x}_{p,i}(t) &= A_{p,i}x_{p,i}(t) + B_{p,i}u_{p,i}(t) \\ y_{p,i}(t) &= C_{p,i}x_{p,i}(t), \end{aligned} \tag{39}$$

where $x_{p,i}(t) \in \mathbb{R}^{n_i}$, $u_{p,i}(t) \in \mathbb{R}^{m_i}$, $y_{p,i}(t) \in \mathbb{R}^{p_i}$ denote the state, input, and output, respectively, of node i for $i = 1, \ldots, N$ and $A_{p,i}, B_{p,i}, C_{p,i}$ are constant matrices with appropriate dimensions. If we define the vectors $x_p(t) = [x_{p,1}(t)^T, \ldots, x_{p,N}(t)^T]^T$, $u_p(t) = [u_{p,1}(t)^T, \ldots, u_{p,N}(t)^T]^T$, and $y_p(t) = [y_{p,1}(t)^T, \ldots, y_{p,N}(t)^T]^T$ we can write (39) in a matrix form as:

$$\begin{aligned} \dot{x}_p(t) &= A_p x_p(t) + B_p u_p(t) \\ y_p(t) &= C_p x_p(t), \end{aligned} \tag{40}$$

where, $x_p(t) \in \mathbb{R}^n$, $u_p(t) \in \mathbb{R}^m$, $y_p(t) \in \mathbb{R}^p$, $n = \sum_{i=1}^N n_i$, $m = \sum_{i=1}^N m_i$, $p = \sum_{i=1}^N p_i$, $A_p = \text{diag}(A_{p,i})$, $B_p = \text{diag}(B_{p,i})$, and $C_p = \text{diag}(C_{p,i})$.

Note that the input vector into the nodes $u_p(t)$ (see Fig. 11) is given by:

$$u_p(t) = u(t)^{in} + u^e(t), \tag{41}$$

where $u^{in}(t) = [u_1^{in}(t)^T, ..., u_N^{in}(t)^T]^T$ is the input vector from the environment to the nodes in the dynamic network and $u^e(t) = [u_1^e(t)^T, \ldots, u_N^e(t)^T]^T$ is the input vector from the dynamic topology (edges) to the nodes. Assume that the interconnections (edges) of the dynamic topology are modeled by transfer functions $e_{ij}(s) = \lambda_{ij}(s)$. Thus, the dynamic consensus protocol that describes the dynamic topology is given in the frequency domain by the following equations:

$$u_i^e(s) = \sum_{j \in \mathcal{N}_{p,i}} \left[\lambda_{ij}(s)(y_{p,j}(s) - y_{p,i}(s))\right], \forall i = 1 \ldots N. \tag{42}$$

Note that the outgoing convention has been used for describing the dynamic topology. The dynamic consensus protocol (42) can be expressed in a matrix form as:

$$u^e(s) = -L^{out}(s)y_p(s), \tag{43}$$

where $L^{out}(s)$ is the outgoing, dynamic Laplacian matrix defined in (26) as

$$L^{out}(s) = B_D^{out}(s)B_S^{out^T} + B_D^{out}(s)B_S^{in^T} = B_D^{out}(s)B_S^T, \tag{44}$$

$$\underbrace{\begin{bmatrix} sC_i^r B_{ij}(s) + A_{ij}(s) & -D_{ij}(s) & -B_{ij}(s) & 0 \\ -D_{ij}(s) & sC_j^r B_{ij}(s) + A_{ji}(s) & 0 & -B_{ij}(s) \end{bmatrix}}_{M(s)|_{s=\frac{d}{dt}}} \begin{bmatrix} T_i \\ T_j \\ Q_i^{in} \\ Q_j^{in} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix},$$

$$M\left(\frac{d}{dt}\right) x(t) = 0. \tag{35}$$

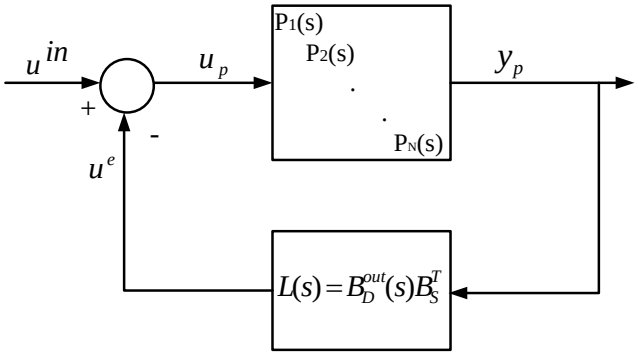

Fig. 11. Block diagram of the dynamic network with heterogeneous nodes and dynamic edges.

where $B_D^{out}(s)$ and B_S are the dynamic outgoing incident matrix and dynamic incident matrix, respectively, and these matrices are defined in Section III for the example shown in Fig. 8.

Combining (43) and (44), we get

$$u^e(s) = -B_D^{out}(s) B_S^T y_p(s), \tag{45}$$

where $B_D^{out}(s)$ matrix captures the dynamics of the edges in the dynamic topology (each column in $B_D^{out}(s)$ describes the dynamic of an edge $e_{ij}(s) = \lambda_{ij}(s)$) and B_S matrix captures the interconnections between the nodes and the edges in the dynamic network.

If the $B_D^{out}(s)$ matrix has a state space realization (A_e, B_e, C_e, D_e), the dynamic topology model (45) can be written in the time domain as:

$$\begin{aligned} \dot{x}_e(t) &= A_e x_e(t) - B_e B_S^T y_p(t) \\ u^e(t) &= C_e x_e(t) - D_e B_S^T y_p(t), \end{aligned} \tag{46}$$

where $x_e(t) \in \mathbb{R}^{\bar{n}}$ is the state vector of the matrix B_S and A_e, B_e, C_e, D_e are constant matrices with appropriate dimensions.

Combining (40), (41) and (46), the overall representation of the dynamic consensus network with N (LTI) heterogeneous nodes and dynamic edges can be written as:

$$\begin{bmatrix} \dot{x}_p(t) \\ \dot{x}_e(t) \end{bmatrix} = \begin{bmatrix} A_p - B_p D_e B_S^T C_p & B_p C_e \\ -B_e B_S^T C_p & A_e \end{bmatrix} \begin{bmatrix} x_p(t) \\ x_e(t) \end{bmatrix} + \begin{bmatrix} B_p \\ 0 \end{bmatrix} u^{in}(t),$$
$$y(t) = \begin{bmatrix} C_p & 0 \end{bmatrix} \begin{bmatrix} x_p(t) \\ x_e(t) \end{bmatrix}. \tag{47}$$

Define $\widetilde{D_e} = D_e B_S^T$ and $\widetilde{B_e} = B_e B_S^T$, the system (47) can be rewritten as

$$\begin{bmatrix} \dot{x}_p(t) \\ \dot{x}_e(t) \end{bmatrix} = \begin{bmatrix} A_p - B_p \widetilde{D_e} C_p & B_p C_e \\ -\widetilde{B_e} C_p & A_e \end{bmatrix} \begin{bmatrix} x_p(t) \\ x_e(t) \end{bmatrix} + \begin{bmatrix} B_p \\ 0 \end{bmatrix} u^{in}(t),$$
$$y(t) = \begin{bmatrix} C_p & 0 \end{bmatrix} \begin{bmatrix} x_p(t) \\ x_e(t) \end{bmatrix}. \tag{48}$$

From the overall representation of the dynamic network (48), the kernel representation of this system can be expressed by:

$$\begin{bmatrix} sI_n - A_p + B_p \widetilde{D_e} C_p & -B_p C_e & -B_p \\ \widetilde{B_e} C_p & sI_{\bar{n}} - A_e & 0 \end{bmatrix} \begin{bmatrix} x_p(t) \\ x_e(t) \\ u^{in}(t) \end{bmatrix} = 0, \tag{49}$$

Using the above kernel representation (49), the following theorem is a direct result of applying the Theorem 4.1 above:

Theorem 5.1: The dynamic consensus network (47) is controllable if and only if the rank of $\begin{bmatrix} sI_n - A_p + B_p \widetilde{D_e} C_p & -B_p C_e & -B_p \\ \widetilde{B_e} C_p & sI_{\bar{n}} - A_e & 0 \end{bmatrix}$ is the same for all $s \in \mathbb{C}$.

A proof of this theorem can be done directly. Alternately, we can use Theorem 9.4 in [20], which states that for two systems S_1 and S_2 connected in feedback configuration shown in Fig. 11, the closed-loop system is controllable if S_1, S_2, and the series connection $S_1 S_2$ are each controllable. The latter condition requires that no pole of S_2 is a zero of S_1. Using this result, we can state the following:

Theorem 5.2: The dynamic network described by (47) is controllable if each of the following is true:

1) The system of nodes defined by (40) is controllable.
2) The system of edges defined by (46) is controllable.
3) No pole of the edge system (46) is a transmission zero [1] of the node system (40).

Proof:

From Theorem

For $s \in \lambda(A_e)$: We can rewrite $\bar{Q}(s)$ in (**??**) as

$$\bar{Q}(s) = \bar{Q}_1(s) \bar{Q}_2(s), \tag{50}$$

where

$$\bar{Q}_1(s) = \begin{bmatrix} I_n & 0 & 0 \\ 0 & \widetilde{B_e} & sI_{\bar{n}} - A_e \end{bmatrix},$$
$$\bar{Q}_2(s) = \begin{bmatrix} sI_n - A_p & -B_p & 0 \\ C_p & 0 & 0 \\ 0 & 0 & I_{\bar{n}} \end{bmatrix}. \tag{51}$$

Form (51), condition 2. in this theorem implies that the $rank(sI_{\bar{n}} - A_e, \widetilde{B_e}) = \bar{n}$, for all s and then $rank(\bar{Q}_1(s)) = n + \bar{n}$, for all s. For $s \in \lambda(A_e)$, the third condition implies that the $rank(\bar{Q}_2(s)) = n + p + \bar{n}$, for all $s \in \lambda(A_e)$, where $p = \sum_i p_i$ is the dimension of the output vector $y_p(t)$ of the system.

Applying Sylvester's rank inequality (see [21] and [22]) in the equation (50), we get

$$n + \bar{n} \geq rank(\bar{Q}(s)) \geq rank(\bar{Q}_1(s)) + rank(\bar{Q}_2(s)) - d, \tag{52}$$

where $d = n + p + \bar{n}$ is the number of columns in the matrix $\bar{Q}_1(s)$.

From (52) and (51), we can rewrite the last inequality as

$$n + \bar{n} \geq rank(\bar{Q}(s)) \geq (n + \bar{n}) + (n + p + \bar{n}) - (n + p + \bar{n}). \tag{53}$$

[1] A transmission zero for $\dot{x} = Ax + Bu$, $y = Cx$ is a value $\lambda \in \mathbb{C}$ such that $\begin{bmatrix} \lambda I_n - A & B \\ C & 0 \end{bmatrix}$ loses rank. Transmission zeros correspond to terms $e^{\lambda t}$ that will not appear in the output when they are in the input.

From the above equation we can conclude that the $rank(\bar{Q}(s)) = n + \bar{n}$, for all $s \in \lambda(A_e)$ and then the rank of the matrix $Q(s)$ is the same for all s and hence the system is controllable. ■

Note that Theorem 5.1 and Theorem 5.2 are equivalent. The controllability conditions introduced in Theorem 5.2 are attractive because it concludes that in addition to the controllability of the nodes and edges, controllability requires the poles of the edges not to match the transmission zeros of the nodes. Furthermore, the third condition can be investigated using only the region of the eigenvalues of the system matrix A_e. This makes controllability analysis more straightforward because it can be analyzed using only the bounds of $\lambda_i(A_e)$ or by simply checking the poles and zeros of the nodes and edges dynamics.

VI. Conclusion

This paper studied a generalization of consensus network problems whereby the network edges' weights are no longer modeled as static gains. Instead, they are represented as dynamic systems coupling the nodes. We call such networks dynamic consensus networks because, under some conditions, all node variables converge to a common value called a consensus. We presented examples of how dynamic graphs can arise in applications. Detailed studies of modeling thermal processes in buildings as directed dynamic graphs were presented. Motivated by these examples, a framework was proposed 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. The dynamic consensus networks addressed herein considered various dynamics of nodes and interconnection topology, including (1) directed dynamic networks with integrator nodes and real-rational transfer function edges; (2) undirected dynamic networks with integrator nodes and strictly-positive-real transfer function edges; and (3) undirected dynamic networks with identical linear time-invariant nodes and dynamic edges. We used the established aspects and properties of the defined dynamic graph theory in conjunction with the behavioral approach to developing a controllability-analysis methodology for dynamic networks. Additionally, the dynamic interconnection topology application in analyzing the interconnections of nodes to establish controllability analysis was demonstrated. The behavioral approach was described, followed by a discussion of its use in the two-rooms example's controllability. This led to presenting the resulting general usage of dynamic graph theory in conjunction with the behavioral approach for controllability-analysis of dynamic consensus networks.

Acknowledgments

The authors would like to thank Professor Kevin L. Moore and Professor Tyrone L. Vincent at the Colorado School of Mines for their help and advice.

References

[1] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Transactions on automatic control*, vol. 49, no. 9, pp. 1520–1533, 2004.

[2] 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*, vol. 18, no. 1, pp. 3078–3083, 2011.

[3] Fadel Lashhab, "Estimation of Dynamic Laplacian Eigenvalues in Dynamic Consensus Networks," *IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS)*, pp. 1–10, 2020.

[4] Fadel Lashhab, "Dynamic Consensus Networks: Dynamic Graph Definitions and Controllability Analysis Using the Behavioral Approach," *IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS)*, pp. 1–9, 2020.

[5] Fadel Lashhab, Kevin Moore, Tyrone Vincent, Deyuan Meng, and Khalid Kuwairi, "Robust H_∞ controller design for dynamic consensus networks", International Journal of Control, vol. 91, no. 8, pp. 1906–1919, 2018.

[6] D.B. Crawley, L.K. Lawrie, F.C. Winkelmann, WF Buhl, Y.J. Huang, C.O. Pedersen, R.K. Strand, R.J. Liesen, D.E. Fisher, M.J. Witte, et al. EnergyPlus: creating a new-generation building energy simulation program. *Energy & Buildings*, 33(4):319–331, 2001.

[7] J.E. Braun and N. Chaturvedi. An inverse gray-box model for transient building load prediction. *HVAR&R Research*, 8(1):73–100, 2002.

[8] K. Lee and J.E. Braun. Model-based demand-limiting control of building thermal mass. *Building and Environment*, 43:1633–1646, 2008.

[9] X. Xu and S. Wang. Optimal simplified thermal models of building envelope based on frequency domain regression using genetic algorithm. *Energy and Buildings*, 39:525–536, 2007.

[10] K.-K. Oh, F. Lashhab, K. L. Moore, T. L. Vincent, and H.-S. Ahn, "Consensus of positive real systems cascaded with a single integrator," *International Journal of Robust and Nonlinear Control*, vol. 25, no. 3, pp. 418–429, 2015.

[11] X. Liu, H. Lin, and B. Chen, "A graph-theoretic characterization of structural controllability for the multi-agent system with switching topology," in *Proceedings of the 48th IEEE Conference on Decision and Control, held jointly with the 28th Chinese Control Conference*, pp. 7012–7017, IEEE, 2009.

[12] H. Tanner, "On the controllability of nearest neighbor interconnections," in *Proceedings of the 43rd IEEE Conference on Decision and Control*, vol. 3, pp. 2467–2472, IEEE, 2004.

[13] M. Ji, A. Muhammad, and M. Egerstedt, "Leader-based multi-agent coordination: Controllability and optimal control," in *Proceedings of 2006 American Control Conference*, pp. 1358–1363, IEEE, 2006.

[14] A. Rahmani, M. Ji, M. Mesbahi, and M. Egerstedt, "Controllability of multi-agent systems from a graph-theoretic perspective," *SIAM Journal on Control and Optimization*, vol. 48, no. 1, pp. 162–186, 2009.

[15] J. Willems, "The behavioral approach to open and interconnected systems," *IEEE Control Systems Magazine*, vol. 27, no. 6, pp. 46–99, 2007.

[16] J. Willems, "On interconnections, control, and feedback," *IEEE Transactions on Automatic Control*, vol. 42, no. 3, pp. 326–339, 1997.

[17] J. Willems and Y. Yamamoto, "Linear differential behaviors described by rational symbols," in *Proceedings of the 17th IFAC World Congress, Seoul, Korea*, pp. 12266–12272, 2008.

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

[19] J. Willems, "The Behavioral Approach to Open and Interconnected Systems Modeling by tearing, zooming, and linking," *IEEE Control Systems Magazine*, vol. 27, no. 6, pp. 46–99, 2007.

[20] C. Chen, *Linear System Theory and Design*. Oxford University Press, Inc., 1998.

[21] G. Matsaglia and G. Styan, "Equalities and inequalities for ranks of matrices†," *Linear and Multilinear Algebra*, vol. 2, no. 3, pp. 269–292, 1974.

[22] A. Ramakrishna and N. Viswanadham, "Decentralized control of interconnected dynamical systems," *IEEE Transactions on Automatic Control*, vol. 27, no. 1, pp. 159–164, 1982.

[23] Vikas Kumar Mishra, Ivan Markovsky, and Ben Grossmann, "Data-Driven Tests for Controllability,"*IEEE Control Systems Letters*, vol. 5, no. 2, pp. 517–522, 2020.

[24] M. Ji and M. Egersted, "A graph-theoretic characterization of controllability for multi-agent systems,"*In IEE American Control Conference*, ACC'07, pp. 4588–4593., 2007.

[25] M. Ji, A. Muhammad, and M. Egerstedt, "Leader-based multi-agent coordination: Controllability and optimal control,"In IEEE American Control Conference, pp. 6, 2006.

[26] A. Rahmani and M. Mesbahi, "On the controlled agreement problem,"*In IEEE American Control Conference*, pp. 6, 2006.

[27] Amirreza Rahmani, Meng Ji, Mehran Mesbahi, and Magnus Egerstedt, "Controllability of multi-agent systems from a graph-theoretic perspective,"*SIAM Journal on Control and Optimization*, vol. 48, no. 1, pp. 162–186, 2009.

[28] M. Zamani and H. Lin, "Structural controllability of multi-agent systems,"*In IEEE American Control Conference*, ACC'09, pp. 5743–5748, 2009.

[29] Tadeusz Kaczorek "Minimum energy control of positive continuous-time linear systems with bounded inputs,"*International Journal of Applied Mathematics and Computer Science*, vol. 23, no. 4, pp. 725–730, 2013.

[30] Jerzy Klamka "Controllability and minimum energy control problem of fractional discrete-time systems," *In New Trends in Nanotechnology and Fractional Calculus Applications*, pp. 503–509, Springer, 2010.

[31] He, Wangli and Xu, Bin and Han, Qing-Long and Qian, Feng," *Adaptive consensus control of linear multiagent systems with dynamic event-*

triggered strategies, IEEE transactions on cybernetics, vol. 50, no. 7, pp. 2996–3008, 2019.

[32] Ren, Wei, "*Multi-vehicle consensus with a time-varying reference state*, Systems & Control Letters, vol. 56, no. 7-8, pp. 474–483, 2007.