

# Consensus Issues in Multi-Agent-Based Distributed Control with Communication Link Impairments



O. S. Akinwale<sup>1\*</sup>, D. F. Mojisola<sup>2</sup>, P. A. Adediran<sup>3</sup>

<sup>1</sup>Electrical and Electronics Engineering Department University of Ilorin, Ilorin Kwara State, Nigeria.

<sup>2</sup>Computer Engineering Department, The Federal University of Technology, Akure, Nigeria.

<sup>3</sup>Electrical and Electronics Engineering Department, The Federal University of Technology, Akure, Nigeria.



**ABSTRACT:** In multi-agent systems, achieving consensus among autonomous agents is a fundamental problem with wide-ranging applications, from autonomous robotics to distributed sensor networks. However, the real-world deployment of such systems often involves communication links prone to impairments, including packet loss, delays, and network congestion. These communication challenges present formidable obstacles to achieving consensus reliably and efficiently. In this paper, consensus protocols were introduced for network with and without communication impairments and convergence analysis were provided in all the cases. The intricate dynamics of consensus issues in multi-agent-based distributed control under the influence of communication link impairments, connectivity and consensus protocol were established. Undirected communication graphs used to model the topology for agents' connectivity is significant to addressing consensus issues of communicating agents. The paper also discusses the tradeoffs and design considerations in developing consensus strategies resilient to communication failures while optimizing performance. Simulation results show that an isolated agent in a network can achieve consensus only when there is a reference value. It was also established that communication impairments significantly degrade the performance of distributed agents in a network.

**KEYWORDS:** consensus protocol, multi-agent systems, distributed control, link impairments, communication graph.

[Received July 12, 2023; Revised Jan 24, 2024; Accepted Feb 1, 2024]

Print ISSN: 0189-9546 | Online ISSN: 2437-2110

## I. INTRODUCTION

In distributed control systems, where multiple autonomous agents collaborate to achieve a common objective, achieving consensus is a fundamental challenge. In this context, consensus refers to the process by which these agents coordinate their actions or states to reach an agreement, even in communication link impairments. The convergence of distributed agents to a shared decision or state is critical in applications ranging from autonomous robotics, coordination control of multi-microgrid (Han *et al.*, 2017; Han *et al.*, 2018; Liu *et al.*, 2018), distributed coordination of collision avoidance of multiple ships (Li *et al.*, 2019), cooperative control of drones (Yanmaz *et al.*, 2018), sensor network for crop irrigation monitoring, traffic light monitoring, intrusion detection schemes (Cruz-piris, 2018; Villarrubia *et al.*, 2017; Xianji *et al.*, 2017)—and sensor networks to multi-agent systems in finance.

One of the main complications of the distributed consensus problem is the real-world limitations of communication channels. Agents in a multi-agent system communicate with each other through network links, and these links are susceptible to various impairments, such as packet loss, delays, and network congestion. These impairments can lead to

communication failures, making it challenging for agents to reach a consensus efficiently.

The study of consensus issues in multi-agent-based distributed control with communication link impairments has gained significant attention in recent years (Giannini *et al.*, 2016; Shrivastava *et al.*, 2019). Researchers and engineers are exploring novel algorithms, protocols, and strategies to address the unique challenges posed by unreliable communication channels. These challenges include ensuring robustness against communication failures, maintaining consensus in network disruptions, and optimizing the tradeoff between communication overhead and consensus performance. Han *et al.*, (2017) introduced both a consensus-oriented distributed coordination controller and a containment controller. These controllers were devised to address voltage regulation and the management of reactive power-sharing regulations within AC microgrids. In, (Shang, 2020) the consensus protocol constrains the agent's state to a predefined projected boundary to deal with uncooperative, anonymous agents in the network of multi-agent systems capable of causing misinformation. The fault-tolerant consensus algorithm for multi-agent systems was investigated by (Wang and Wang, 2018).

In leaderless and leader-follower communication structures, fully distributed adaptive protocols were designed for agents with nonlinear dynamics. The proposed protocol is independent of the Laplacian matrix eigenvalues linked with

\*Corresponding author: olayanju.sa@unilorin.edu.ng

doi: <http://dx.doi.org/10.4314/njtd.v21i1.2212>

the communication graph, thereby fully distributing the implemented protocol. To provide distributed consensus protocols for multi-agent systems with common continuous-time linear dynamics, (Li and Duan, 2014) present a consensus region technique. The design of the feedback gain matrices for the consensus protocols is decoupled from the communication graph using a beneficial decoupling feature of the consensus region approach. Distributed adaptive protocols with time-varying coupling weights are established for applications with undirected and directed graphs, and they can be fully distributed to overcome this restriction.

The task of designing a completely distributed output feedback consensus procedure for multi-agent systems with general linear dynamics over directed graphs is the subject of the work in (Lv *et al.*, 2018). For tightly connected directed graphs or leader-follower graphs, the authors suggested innovative distributed observer-based adaptive consensus methods instead of the current consensus techniques for adaptive output feedback.

This paper delves into the key issues and solutions to consensus problems in multi-agent-based distributed control with communication link impairments. The work explores the theoretical foundations of consensus algorithms, examines practical scenarios where these algorithms are applied, and discusses the tradeoffs in designing resilient and efficient consensus strategies for networked multi-agent systems. Addressing these challenges aims to shed light on the intricate interplay between distributed control, communication impairments, and the pursuit of consensus in a world increasingly reliant on interconnected and collaborative autonomous agents.

## II. FUNDAMENTALS OF GRAPH THEORY

Directed graphs (digraphs) or undirected graphs can be used for multi-agent topology representation to achieve distributed control among agents. It describes the topology between agents in a network. Given  $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$  is a weighted digraph of order  $n$ , having the set of nodes  $\mathcal{V} = \{\mathcal{V}_1, \dots, \mathcal{V}_n\}$ , set of edges  $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ , and a weighted adjacency matrix  $\mathcal{A} = [a_{ij}]$  where  $a_{ij}$  are non-negative adjacency elements. The nodes of the digraph represent each agent in a network. The edges denote communication links between connected nodes. Each edge  $\mathcal{E}_{ij} = (\mathcal{V}_i, \mathcal{V}_j)$  implies that agent  $i$  obtains information from agent  $j$ . (Olfati-saber and Murray, 2004; Zhang *et al.*, 2017). For a digraph, an agent  $i$  only receives information from its neighbours  $N_i$ ,  $a_{ij}$  is the weight of edge  $\mathcal{E}_{ij} = (\mathcal{V}_i, \mathcal{V}_j)$ , and  $a_{ij} = 1$  if  $(\mathcal{V}_i, \mathcal{V}_j) \in \mathcal{E}$ , otherwise  $a_{ij} = 0$ . The set of neighbours of node  $\mathcal{V}_i$  is denoted by:

$$N_i = \{\mathcal{V}_j \in \mathcal{V} : (\mathcal{V}_i, \mathcal{V}_j) \in \mathcal{E}\} \quad (1)$$

The in-degree matrix is defined as  $\mathcal{D} = \text{diag}\{d_i\} \in R^{N \times N}$  with  $d_i = \sum_{j \in N_i} a_{ij}$ . The Laplacian matrix is defined as  $\mathcal{L} = \mathcal{D} - \mathcal{A}$ . A direct path linking node  $i$  to node  $j$  is a sequence of edges, expressed as  $\{(\mathcal{V}_i, \mathcal{V}_k), (\mathcal{V}_k, \mathcal{V}_l), \dots, (\mathcal{V}_m, \mathcal{V}_j)\}$ . A digraph is said to have a minimum spanning tree (MST), if

there is a root node with a direct path linking the node to every other node in the communication graph.

## III. CONTINUOUS-TIME CONSENSUS

In a multi-agent system comprising of  $N$  agents with continuous-time linearized dynamics. The system dynamics is represented by;

$$\dot{x}_i = Ax_i + Bu_i \quad (2)$$

$$y_i = Cx_i, i = 1, \dots, N \quad (3)$$

Where  $A$ ,  $B$ , and  $C$  are constant matrices with dimensions proportional to the number of agents,  $N$ , in the network. In continuous-time consensus, each agent communicates and shares information with only its neighbors. A fixed number of  $N$ -agents attain consensus when  $\lim_{t \rightarrow \infty} \|x_i(t) - x_j(t)\| = 0, \forall i, j = 1, \dots, N$ . The fundamental goal of adopting the consensus algorithm is to achieve collective agreement by applying the agents' dynamics to the information collected from each agent. Section II describes the graph theories essential for realizing the consensus algorithm. If the dynamics of each agent are represented by  $\dot{x}(t) = u_i(t)$ , the network is represented by a time-varying communication graph  $\mathcal{G}(t) = (\mathcal{V}, \mathcal{E}(t))$ , where  $\mathcal{V} = \{1, 2, \dots, n\}$  denotes the number of agents, and  $\mathcal{E}(t)$  is the time-varying edge connecting the agents. For this paper, all agents are in the graph,  $\mathcal{G}$ , the scenarios when the topology is fixed, with and without communication link impairments, were considered. Local control protocol for  $i$ th agent is given for fixed topology without communication delay by (Lewis *et al.*, 2014; Olfati-Saber and Murray, 2004):

$$u_i(t) = \sum_{j \in N_i} a_{ij} [x_j(t) - x_i(t)] \quad (4)$$

Since.  $\dot{x}(t) = u_i(t)$ , consensus algorithm for linear systems can be represented by:

$$\dot{x}_i(t) = \sum_{j \in N_i} a_{ij} [x_j(t) - x_i(t)] \quad (5)$$

where  $i = \{1, 2, \dots, n\}$  is the index of agents,  $\mathcal{A}(t) = [a_{ij}]$ , the adjacency matrix,  $N_i(t) = \{j \in \mathcal{V} : a_{ij} \neq 0\}$  set of agent  $i$  neighbors,  $x_i(t)$  and  $x_j(t)$  are state information from agents  $i$  and  $j$ , respectively, and  $\dot{x}_i(t)$  is the state of agent  $i$ .

The topology with communication delay for a fixed graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$  and communication delay  $\tau_{ij} > 0$ , corresponding to edge  $\mathcal{E}_{ij} \in \mathcal{E}$ , the linear time-delayed consensus protocol can be expressed as (Nojavanzadeh *et al.*, 2021):

$$u_i(t) = \sum_{j \in N_i} a_{ij} [x_j(t - \tau_{ij}) - x_i(t - \tau_{ij})] \quad (6)$$

## IV. COMMUNICATION LINK IMPAIRMENT MODELLING

This study focuses on the interplay between agents, composed of two key elements: the communication graph and the consensus protocol. The communication graph establishes the communication topology among the agents. It defines how they interact and share information within the network. On the other hand, the consensus protocol governs the relationships between individual agents (communicating nodes) and their neighbors, facilitating the process of reaching a consensus. The

ultimate goal of this collaboration is to achieve synchronization among cooperating nodes, thereby solving the agreement problem between agents. When the agents effectively reach a consensus on the communication graph, it signifies successful synchronization of the interacting agents. Generally, the characteristics of each link between connected nodes can be represented by a stable transfer function  $h(s)$ . In this paper, the link represents a wireless communication channel affected by communication link impairments being studied: a communication link time delay, communication link distortion or noise, and communication link failure.

#### A. Communication Link Time Delay

This section presents the mathematical modelling of communication link time delay. Considering a network with a fixed topology  $G=(V,E,A)$  for sets of nodes where the information of the node state  $\mathcal{V}_j$  is transmitted to a neighboring node  $\mathcal{V}_i$  through a communication link  $\mathcal{E}_{ji}$  of transfer function expressed in Laplace form as:  $h_{ji} = e^{(-\tau_{ij}s)}$ . Applying the linear protocol given in equation 6 and introducing a consensus gain,  $-K_{c,i}$ , the state of the node with delay is expressed as:  $\dot{x}_i(t) = -K_{c,i} \sum_{u_j \in N_i} a_{ij} [x_i(t - \tau_{ij}) - x_j(t - \tau_{ij})]$  (7)

Then, the value of the nodes is the solution of the following linear delay differential equation:

$$\dot{x} = -\mathcal{L}x(t - \tau_{ij}) \quad (8)$$

In this case of delay,  $\tau_{ij}$ , average consensus can be reached globally asymptotically if and only if,  $\tau_{ji} \in (0, \tau_{ji}^*)$  with  $\tau_{ji}^* = \pi/2\lambda_n$ ,  $\lambda_n = \lambda_{max}(\mathcal{L})$ , where  $\lambda_{max}(\mathcal{L})$  is the maximum eigenvalue of the Laplacian matrix.

#### B. Communication Link Noise

The case where the communication link is impaired by noise, given that each node  $\mathcal{V}_i$  of the connected graph  $G$  receives the information  $x_j$  from its neighboring nodes through a noisy link described as  $h(s) = \sigma_i \omega_i$ . The linear consensus protocol becomes:

$$u_i(t) = \sum_{j \in N_i} a_{ij} [x_j(t) - x_i(t)] + \sigma_i \omega_i \quad (9)$$

where  $\sigma_i$  is the noise standard deviation, and  $\omega_i$  is a zero-mean white noise process for the  $i$ th node. The state of node  $\mathcal{V}_i$  after passing through the noisy link is represented by Eqn. 10 where a proportional controller gain,  $-K_{c,i}$ , has been introduced.

$$\dot{x}_i(t) = -K_{c,i} \sum_{j \in N_i} a_{ij} [x_j(t) - x_i(t)] + \sigma_i \omega_i \quad (10)$$

#### C. Communication Link Failure

Communication link failure is when one or more edges connected to a particular node are not communicating; that is, no information is sent or received. Such an edge is considered a failed link. Considering a network with a fixed topology  $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$  for sets of nodes where the information of the node state  $\mathcal{V}_j$  is transmitted to a neighboring node  $\mathcal{V}_i$  through a communication link  $\mathcal{E}_{ji}$ . A node  $\mathcal{V}_i$  only receives information from its neighbors  $N_i$ .  $a_{ij}$  is the weight of the edge  $\mathcal{E}_{ij} = (\mathcal{V}_i, \mathcal{V}_j)$  and  $a_{ij} = 1$  if  $(\mathcal{V}_i, \mathcal{V}_j) \in \mathcal{E}$ , otherwise  $a_{ij} = 0$ . The set of neighbors of a node  $\mathcal{V}_i$  is as denoted by Eqn. 1.

Link failure results when a node,  $\mathcal{V}_j$ , forms an edge with another  $\mathcal{V}_i$  within  $N_i$  i.e.  $(\mathcal{V}_i, \mathcal{V}_j) \in \mathcal{E}$  but  $a_{ij} = 0$ . The weight of its adjacency matrix determines the state of an individual node, which is a combination of link information among all the nodes in the neighborhood of node  $i$ .

## V. SIMULATION OF COMMUNICATION IMPAIRMENTS

Figure 1 shows a communication scenario in a microgrid (MG) with four distributed generators (DGs). In the Figure, the wireless communication graph defines the communication network topology. A node (DG) in the network can only communicate with its neighbors. Each of the DGs shares the values of their output frequency of operation or the node's state during communication.

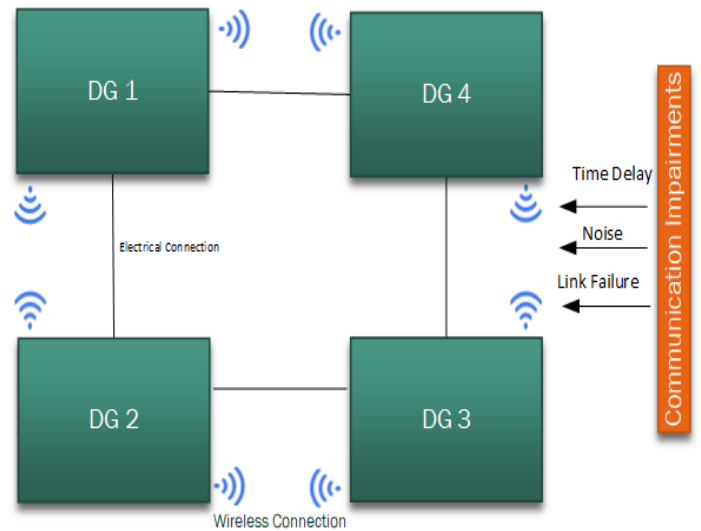


Figure 1: Communication scenario in a Microgrid with four DGs

This section introduces the interactive matrix describing communication among the nodes in the network, visualized through four different communication graphs shown in Figure 2. Four nodes (agents) DG1, DG2, DG3, and DG4 communicate to reach a consensus frequency value on a graph network. In Figure 2a, the graph is fully connected and undirected, enabling bidirectional information exchange among all nodes. Each edge's weight is set to 1, indicating robust connectivity. The corresponding adjacency matrix derives the Laplacian matrix,  $a_{ij}$ , is defined by Eqn. 11a. The second eigenvalue,  $\lambda_2$ , also called the Fiedler eigenvalue is used to quantify the graph's algebraic connectivity and to determine the nodes' consensus convergence rate. As graph connectivity increases, the Fiedler eigenvalue also rises, facilitating faster node convergence.

In Figure 2b, the graph is partially connected, allowing each node to communicate except with a link failure between DG1 and DG4. Figure 2c exhibits link failures that entirely isolate node 4. In Figure 2d, node 1 assumes the role of a leader or reference node, broadcasting information without receiving any from its neighbors. The adjacency matrices for the graphs in Figures 2a to 2d are defined by Eqns. 23a to 23d, respectively.

$$a_{ij1} = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix}, a_{ij2} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}, a_{ij3} = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, a_{ij4} = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} = (11a - d)$$

The eigenvalues of the Laplacian matrices are presented in Table 1. The second eigenvalue  $\lambda_2$  of the graph Laplacian matrix  $\mathcal{L}$  is essential in determining the speed of interaction of dynamic systems on graphs. Graph topologies that have a significant value of  $\lambda_2$  are better for achieving convergence. The second eigenvalue of  $\mathcal{L}$  the Fiedler eigenvalue, has essential and subtle connections to the graph's topology.

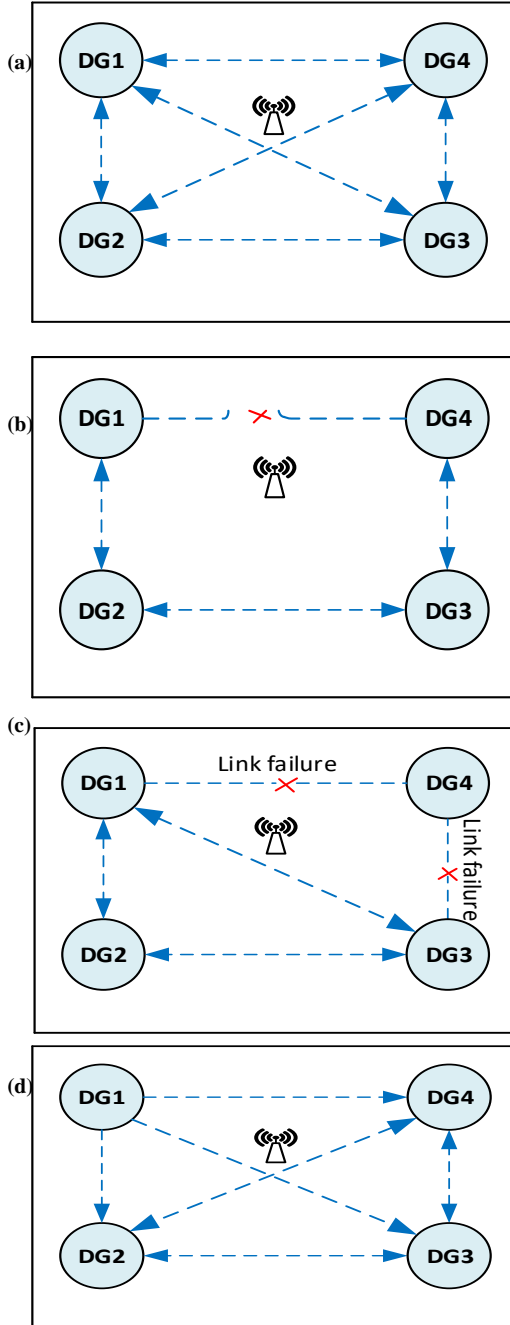


Figure 2: Graph topology for the communication network of a multi-agent system. With: (a) full connectivity (b) partial connectivity (c) connectivity with a link failure (d) connectivity with a leading node DG1

It is also known as the graph algebraic connectivity. The graph in Figure 2a has the highest algebraic connectivity.

## VI. RESULTS AND DISCUSSION

Table 1 summarizes the eigenvalues corresponding to the graph topologies shown in Figures 2 (a-d), and equations 14a to 14d. The time constant ( $\tau$ ) at consensus is  $\tau = \frac{1}{\lambda_2}$ , where  $\lambda_2$  represents the second eigenvalue of the Laplacian matrix,  $\mathcal{L}$ , referred to as the Fiedler eigenvalue. For the communication graphs of Figure 2 with varying connectivity, the respective values of  $\lambda_2$  are: 4, 0.5858, 0 and 4.

Table 1: Eigenvalues of the Laplacian matrices

Eigenvalue	G1	G2	G3	G4
$\lambda_1$	0	0	0	0
$\lambda_2$	4	0.5858	0	4
$\lambda_3$	4	2	3	1
$\lambda_4$	4	3.4142	3	4
$\tau$	0.25	1.7071	inf.	0.25

### A. Consensus Protocol with Communication Link Failure

#### Case 1: Consensus for all DGs with a reference frequency

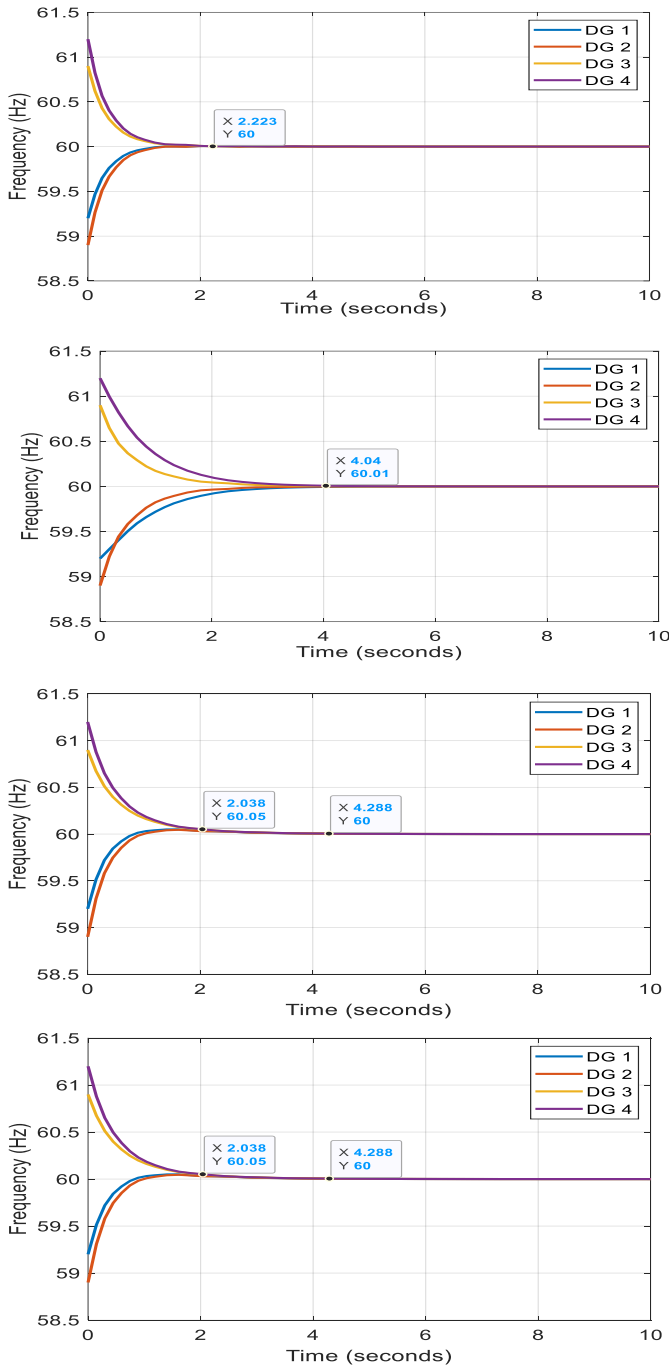
Figure 3 depicts the relationship between connectivity, convergence, and the cost of communication using the communication graph when DG1 is connected to a reference frequency of 60Hz. Graph (a) (Figure 2a) notably exhibits the highest connectivity and a higher connection cost. Furthermore, graph (a) demonstrates the fastest convergence time, resulting in the lowest delay constant, as shown in Figure 3a. As seen in Figure 3b, as connectivity reduces with a link failure between DG1 and DG4, the convergence time significantly increases. However, in the case of graph c (Figure 3c), convergence first took place at 2.11 seconds and at a frequency of 59.97Hz slightly outside the reference frequency of 60Hz.

This is because DG4 is not coordinating with the remaining DGs in the network as a result of link failure, it eventually reaches consensus with the other DGs at a much delayed time of 5.61 seconds. Consensus is made possible only because DG4 is independently connected with the reference frequency. In contrast, graph d features DG1 as the leading node, which only disseminates its generated values without receiving information from other DGs. Consequently, at first, all other DGs synchronize outside the reference frequency, tending towards the frequency of node 1, before eventually reaching consensus at the operating frequency at about 4.2 seconds.

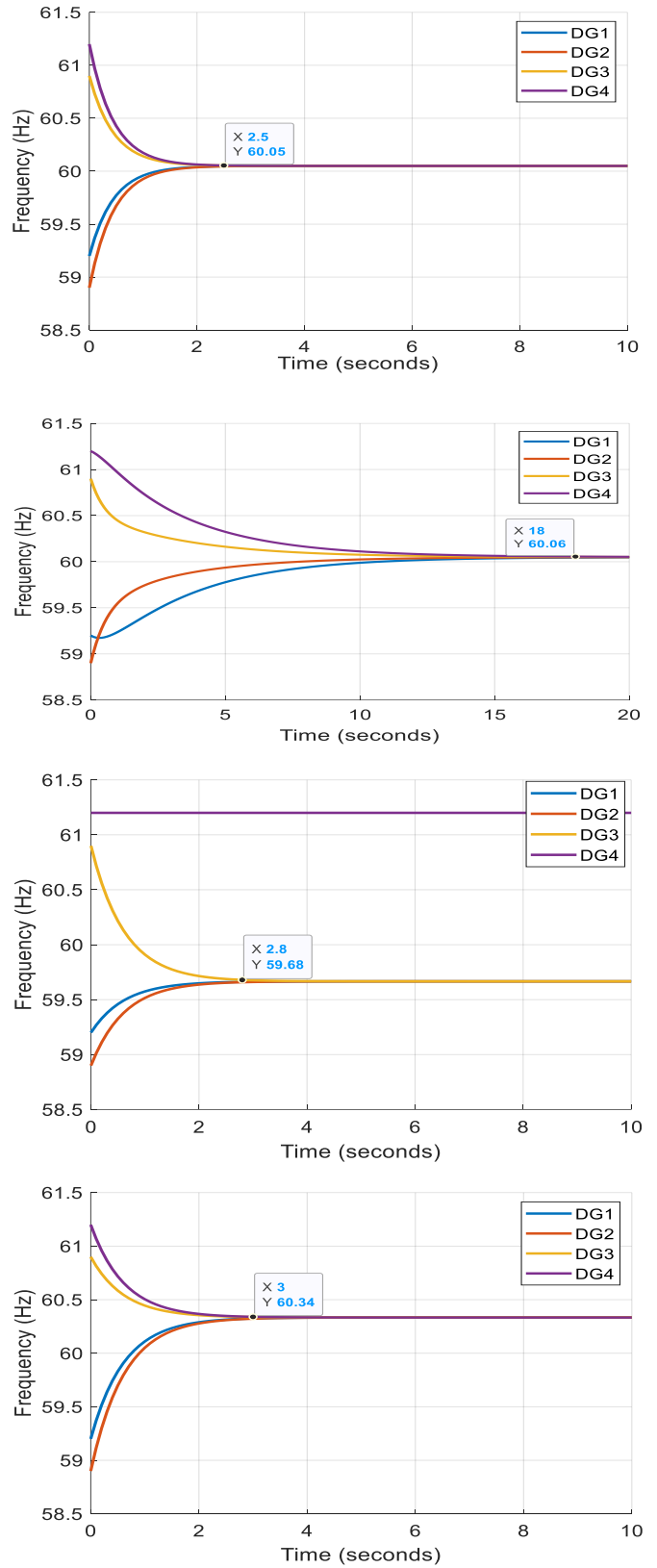
#### Case 2: Consensus for all DGs without reference frequency

Shown in Figure 4 are the relationships between connectivity, convergence, and the cost of communication using the communication graph when there is no reference frequency connected. Comparing the graph in Figure 3a with the graph in Figure 4a, where consensus is reached at 60.05Hz, it became evident that, when there is no reference frequency

connected to any of the DGs, consensus reached is an average value based on the agreement between interacting DGs or nodes. It can also be seen clearly when comparing Figure 3b with Figure 4b, that without a reference signal connected to one of the nodes, the time taken before convergence is finally reached is significantly increased. In the case of Figure 4c, convergence is not possible between all the nodes since one node is not participating in the consensus processes due to a link failure that effectively isolates DG4. Nevertheless, the other three coordinating DGs reached a consensus at a frequency of 59.68 and a time of 2.8 seconds. The graph in Figure 4d when compared with that of Figure 3d, shows consensus takes place outside the operating frequency and at a much delayed time.



**Figure 3:** Effects of connectivity ( $\lambda_2$ ) on agents' consensus with reference frequency connected to DG1: (a):  $\lambda_2 = 4$ , (b):  $\lambda_2 = 0.5858$ , (c):  $\lambda_2 = 0$ , (d):  $\lambda_2 = 4$



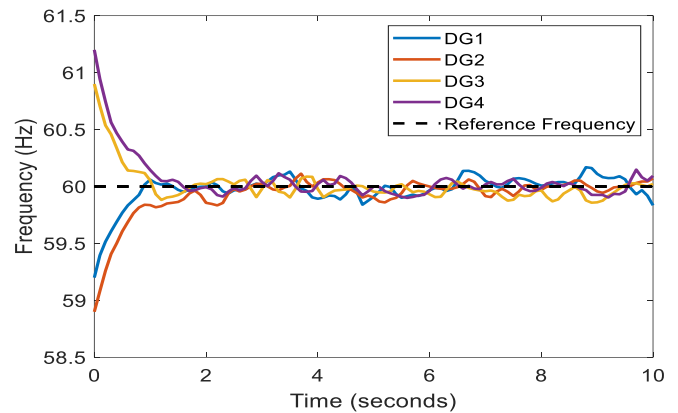
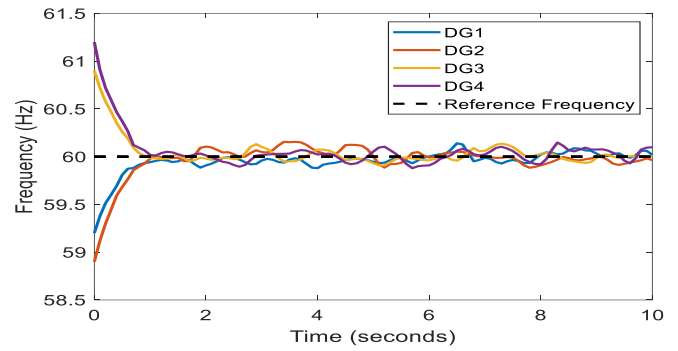
**Figure 4:** Effects of connectivity ( $\lambda_2$ ) on agents' average consensus while communicating on the communication graph without reference frequency: (a):  $\lambda_2 = 4$ , (b):  $\lambda_2 = 0.5858$ , (c):  $\lambda_2 = 0$ , (d):  $\lambda_2 = 4$

**B. Consensus protocol with communication link noise**

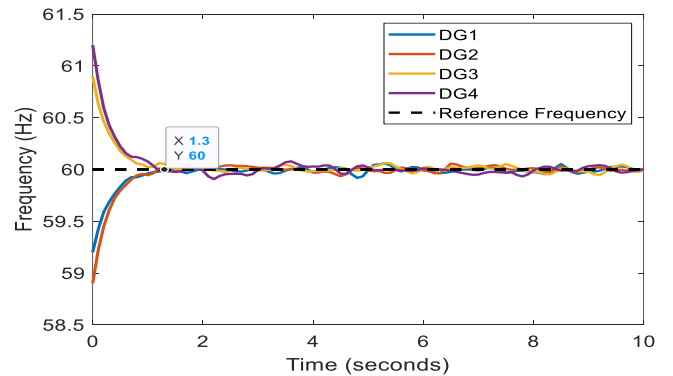
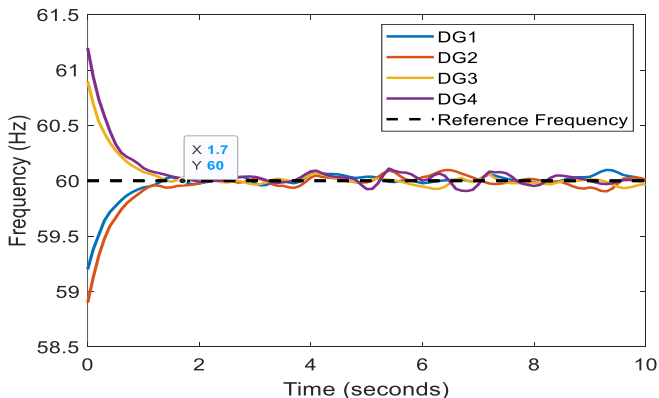
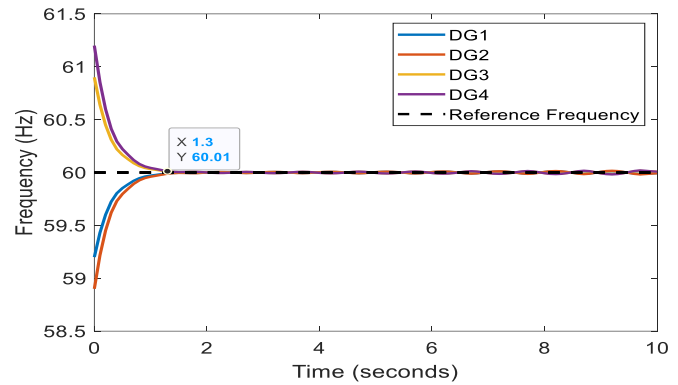
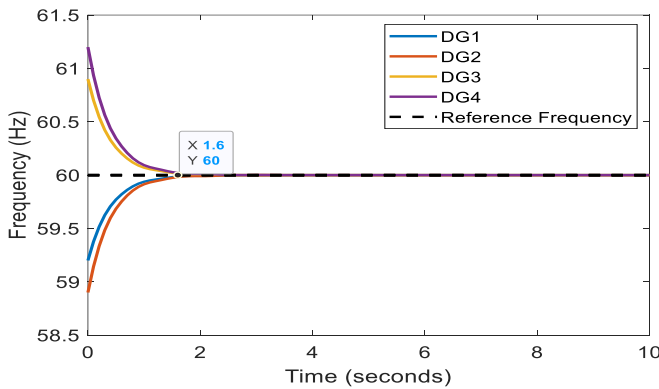
Using the graph in Figure 2 and the corresponding adjacency matrices, the simulations of the consensus equation were conducted with varying noise levels and the following parameters: Graph connectivity ( $\lambda_2$ ) set at 4, and proportional gain ( $K_{c,i}$ ) at 2.5, which is a design parameter. Different noise standard deviations ( $\sigma$ ) were considered: (a)  $\sigma = 0$ , (b)  $\sigma = 0.1$ , (c)  $\sigma = 0.2$ , and (d)  $\sigma = 0.3$ . The reference frequencies for each node were set as follows: node1 at 59.2Hz, node2 at 58.9Hz, node3 at 60.9Hz, and node4 at 61.2Hz. The reference frequency for the system was set to 60Hz.

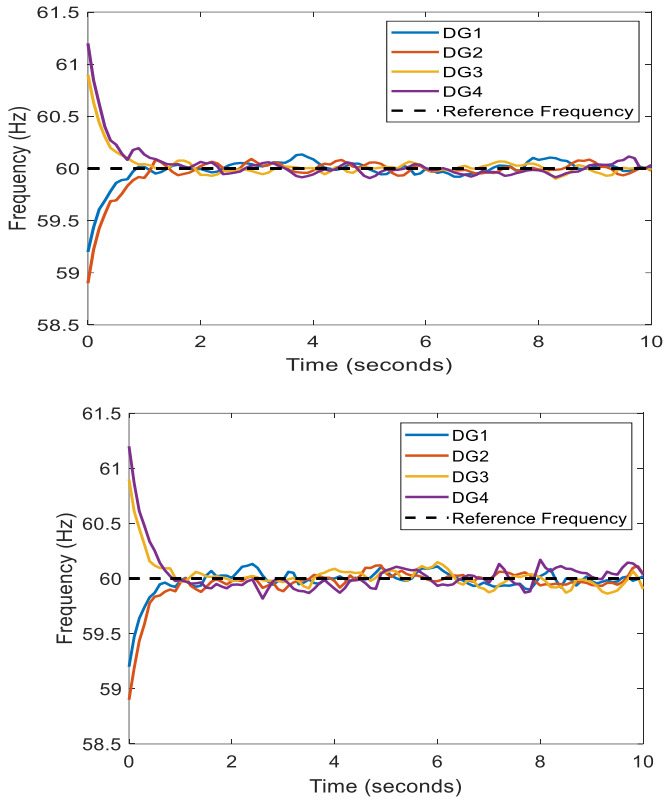
The evaluation focused on assessing the impact of noise on communication within the system network. Figure 5a, with no noise ( $\sigma = 0$ ), exhibited the fastest convergence, with no noticeable distortion as the frequencies tended towards the reference point. Convergence occurred at approximately 1.6 seconds. Figure 5b, with  $\sigma=0.1$ , experienced convergence at around 1.7 seconds, but slight distortion in the frequencies of the nodes was continually observed with time after convergence. However, in Figures 5c and d, with  $\sigma = 0.2$  and  $\sigma = 0.3$ , respectively, noise level hindered apparent convergence in the obtained frequency values.

The impact of controller gain on system convergence is investigated and illustrated in Figure 6. Keeping the same parameters as in the initial case, the controller gain,  $K_{c,i}$ , is now adjusted to 3.4. Notably, a remarkable reduction in the convergence time is observed across all levels of noise standard deviation used in the simulation. Additionally, the distortion introduced by the noise is significantly mitigated.



**Figure 5: Effects of noise on frequency synchronization and stability at  $\lambda_2 = 4$  and  $K_{c,i} = 2.5$ , (a) with  $\sigma = 0$ , (b)  $\sigma = 0.1$ , (c)  $\sigma = 0.2$ , (d)  $\sigma = 0.3 = 4$**



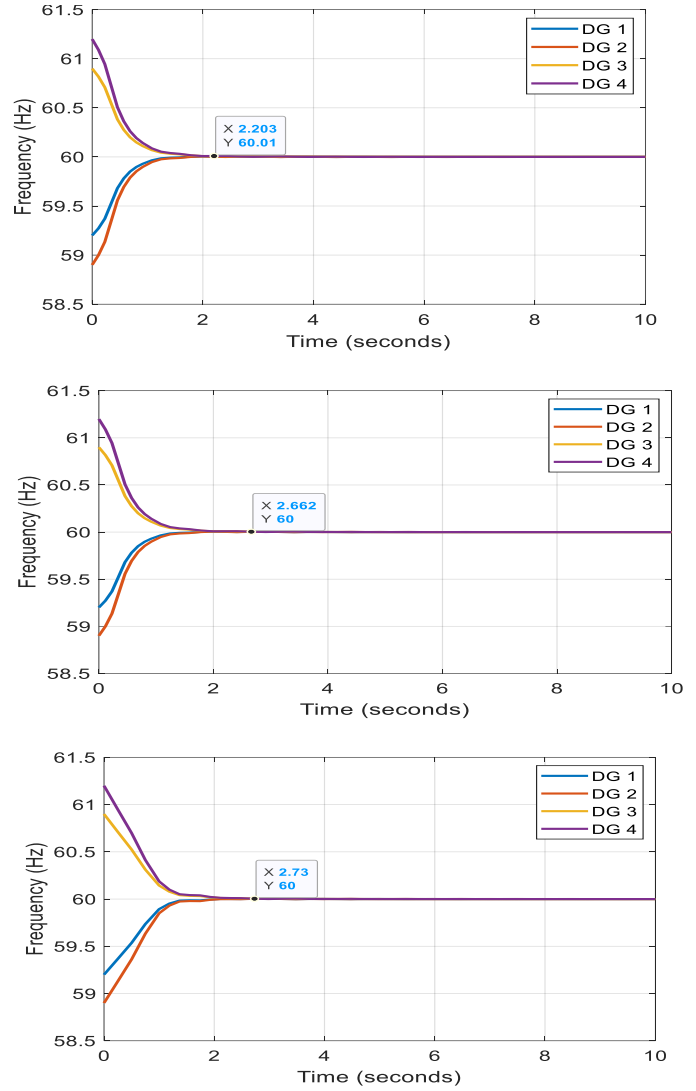
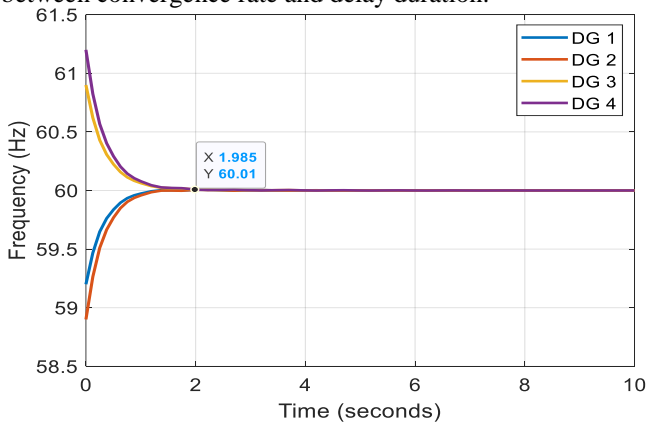


**Figure 6: Effects of noise on frequency synchronization and stability at  $\lambda_2 = 4$  and  $K_{c,i} = 3.4$ , (a) with  $\sigma = 0$ , (b)  $\sigma = 0.1$ , (c)  $\sigma = 0.2$ , (d)  $\sigma = 0.3$**

*C. Effects of communication link time delay on frequency convergence*

*Case 1: Equal communication link time delay among nodes*

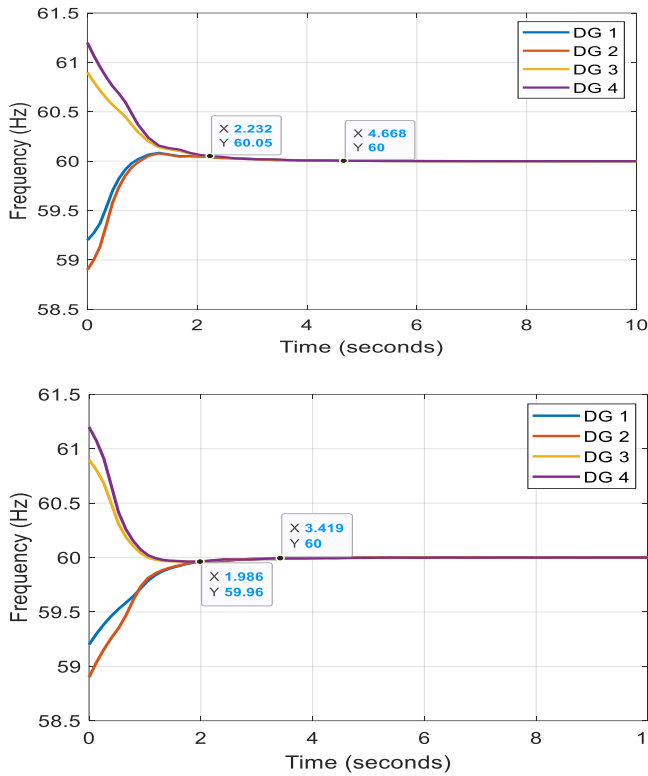
Utilizing Figure 2a alongside Eqn. 11a, which explains the comprehensive algebraic connectivity among all nodes (DGs) within the network, the time delay inherent in communication links between the communicating DGs is uniformly established. The initial delay values are 0, 0.2, 0.4, and 0.6 seconds. A consistent convergence gain,  $K_{c,i}$ , of 0.5 is uniformly applied across all DGs. The impact of communication link time delay on the convergence rate is graphically depicted in Figure 7. It is worth noting that an escalation in delay leads directly to an extension in convergence time. This implies the existence of a tradeoff between convergence rate and delay duration.



**Figure 7: Effects of equal communication time delay on frequency convergence of DGs at  $\lambda_2 = 4$  and  $K_{c,i} = 0.5$ , (a) with  $\tau = 0$ , (b)  $\tau = 0.2$ , (c)  $\tau = 0.4$ , (d)  $\tau = 0.6$**

*Case 2: Varying communication link time delay among nodes*

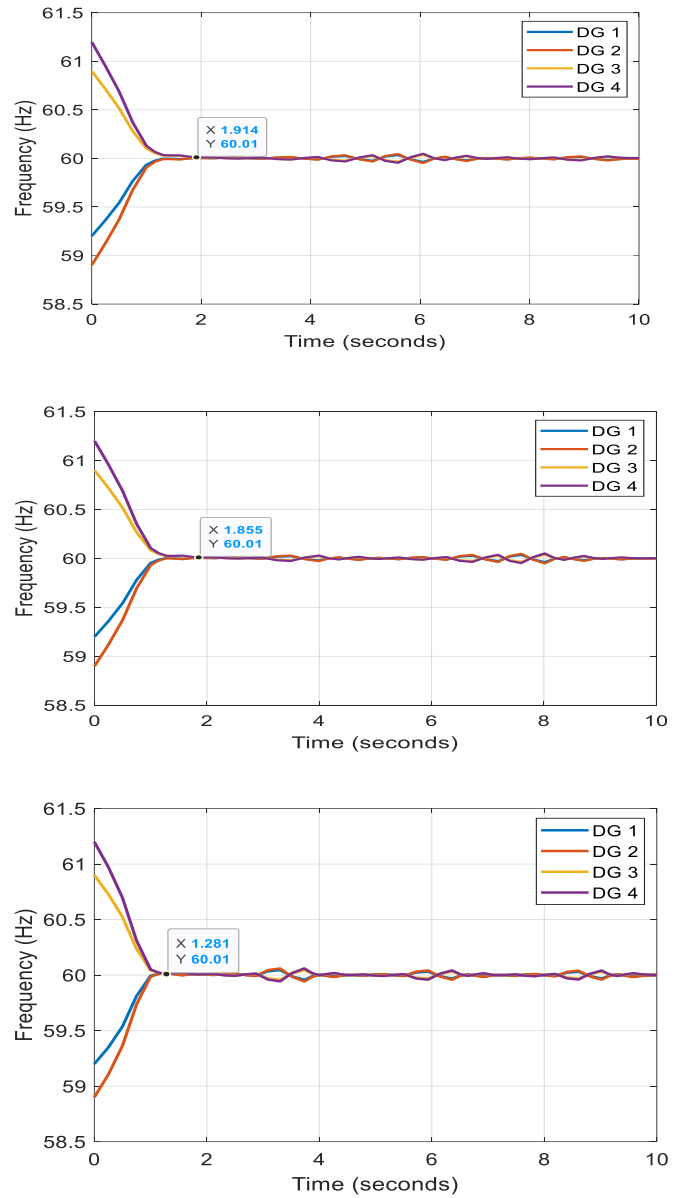
The scenario involving varying communication link time delays among communicating nodes is illustrated in Figure 8. In Figure 8a, communication delays of 0.2 seconds, 0.3 seconds, and 0.6 seconds are imposed between DG1 and DG2, DG1 and DG3, and DG1 and DG4, respectively. In Figure 8b, time delays between DG1 and DG2, DG1 and DG3, and DG1 and DG4 are set at 0.8 seconds, 0.6 seconds, and 0.2 seconds, respectively. Notably, the initial frequency consensus was achieved at 2.232 seconds beyond the reference frequency of 60Hz, followed by consensus at the operational frequency at 4.668 seconds. Upon examining Figures 8a and 8b, it is apparent that initial frequency consensus is attained beyond the reference frequency in the direction of DGs characterized by longer communication link time delays.



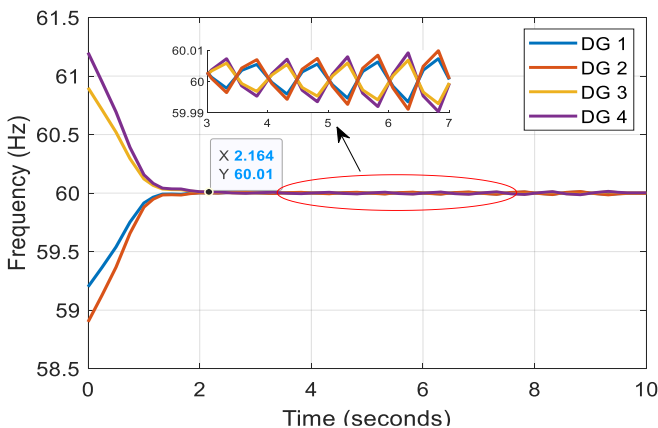
**Figure 8: Effects of variable communication time delay on frequency convergence of DGs at  $\lambda_2 = 4$  and  $K_{c,i} = 0.5$ , (a) with  $\tau_{12} = 0.2, \tau_{13} = 0.3, \tau_{14} = 0.6$  (b)  $\tau_{12} = 0.8, \tau_{13} = 0.6, \tau_{14} = 0.2$**

*Case 3: Effects of Consensus gain on frequency convergence among nodes*

Keeping the communication link time delay of 0.4 seconds constant, the convergence gain is elevated from 0.6 to 1. When juxtaposed with the impact of communication link time delay on the frequency convergence rate, as shown in Figure 7c, a higher convergence gain leads to a decrease in the convergence time. However, it is crucial to acknowledge that surpassing a convergence gain of 0.5 gives rise to frequency distortions even after the convergence point has been attained, as depicted in Figure 9a. The similar outcomes illustrated in Figures 9b, 9c, and 9d underscore the existence of an upper threshold for the convergence gain that can be introduced within the consensus protocol.



**Figure 9: Effects of consensus gain on frequency convergence among DGs at  $\lambda_2 = 4$  and  $\tau = 0.4$ , with (a)  $K_{c,i} = 0.6$ , (b)  $K_{c,i} = 0.7$ , (c)  $K_{c,i} = 0.8$ , (d)  $K_{c,i} = 1$**



**VII. CONCLUSION**

Exploring consensus issues in multi-agent-based distributed control with communication link impairments offers valuable insights into the challenges and opportunities in achieving reliable consensus within networked multi-agent systems. The synthesis of theoretical foundations and practical applications provides a comprehensive perspective on the evolving landscape of distributed control in a world where autonomous agents and unreliable communication are increasingly prevalent. Some of the research findings are summarized as follows:

- (i) In the event of a link failure, an isolated node within the network can still achieve consensus with other nodes, albeit with a significant delay, provided it receives a reference value for consensus. However, when lacking a reference value, attaining consensus with other network nodes becomes an



unattainable goal. If none of the nodes in the network are connected to a reference signal, the consensus algorithm returns an average consensus value among the communicating nodes.

(ii) As the level of noise escalates, the task of synchronizing nodes and preserving precise frequency values becomes progressively more daunting. This underscores the critical need to address communication impairments when designing agents' communication systems. The selection of controller gain emerges as a pivotal factor in fine-tuning the system's performance amid noise.

(iii) It is important to emphasize that an increase in delay directly results in a prolonged convergence time. This highlights the presence of a trade-off between the speed of convergence and the duration of delay.

#### AUTHOR CONTRIBUTIONS

**S.A. Olayanju:** Conceptualization, methodology, literature review, writing and analysis, review and editing. **F. M. Dahunsi and A. A. Ponnle:** Conceptualization, supervision, analysis and revising, review and editing. All authors have read and approved the manuscript.

#### REFERENCES

- Cruz-piris, L. (2018).** *Optimized Sensor Network and Multi-Agent Decision Support for Smart Traffic Light Management*. <https://doi.org/10.3390/s18020435>
- Giannini, S. A. Petitti; D. Di Paola and A. Rizzo. (2016).** Asynchronous Max-Consensus Protocol with Time Delays: Convergence Results and Applications. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 63(2), 256–264. <https://doi.org/10.1109/TCSI.2015.2512721>
- Han, R; L. Meng; G. Ferrari-Trecate; A. A. Coelho; J. C. Vasquez and J. M. Guerrero. (2017).** Containment and Consensus-Based Distributed Coordination Control to Achieve Bounded Voltage and Precise Reactive Power Sharing in Islanded AC Microgrids. *IEEE Transactions on Industry Applications*, 53(6), 5187–5199. <https://doi.org/10.1109/TIA.2017.2733457>
- Han, Y.; K. Zhang; H. Li; E. A. Coelho and J. M. Guerrero. (2018).** MAS-Based Distributed Coordinated Control and Optimization in Microgrid and Microgrid Clusters: A Comprehensive Overview. *IEEE Transactions on Power Electronics*, 33(8), 6488–6508. <https://doi.org/10.1109/TPEL.2017.2761438>
- Lewis, F. L.; Z. Hongwei; H. Kristian and D. Abhijit. (2014).** Cooperative control of multi-agent systems. In *Autonomous Vehicle Navigation*. Springer US. <https://doi.org/10.1201/b19544-7>
- Li, S.; J. Liu and R. R. Negenborn. (2019).** Distributed coordination for collision avoidance of multiple ships considering ship maneuverability. *Ocean Engineering*, 181, 212–226. <https://doi.org/10.1016/j.oceaneng.2019.03.054>
- Li, Z. and Duan, Z. (2014).** Distributed consensus protocol design for general linear multi-agent systems: A consensus region approach. *IET Control Theory and Applications*, 8(18), 2145–2161. <https://doi.org/10.1049/iet-cta.2014.0012>
- Liu, W.; W. Gu; J. Wang; W. Yu and X. Xi. (2018).** Game Theoretic Non-Cooperative Distributed Coordination Control for Multi-Microgrids. *IEEE Transactions on Smart Grid*, 9(November 2019), 6986–6997. <https://doi.org/10.1109/TSG.2018.2846732>
- Lv, Y.; Z. Li and Z. Duan. (2018).** Distributed adaptive consensus protocols for linear multi-agent systems over directed graphs with relative output information. *IET Control Theory and Applications*, 12(5), 613–620. <https://doi.org/10.1049/iet-cta.2017.0615>
- Nojavanzadeh, D.; S. Lotfifard; Z. Liu; A. Saberi; and A. A. Stoorvogel. (2021).** Scale-free Distributed Cooperative Voltage Control of Inverter-based Microgrids with General Time-varying Communication Graphs. *IEEE Transactions on Power Systems*, 1–8.
- Olfati-Saber, R. and Murray, R. M. (2004).** Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control*, 49(9), 1520–1533. <https://doi.org/10.1109/TAC.2004.834113>
- Shang, Y. (2020).** Resilient consensus in multi-agent systems with state constraints. *Automatica*, 122(ISSN 0005-1098), 1–9.
- Shrivastava, S.; B. Subudhi and S. Das. (2019).** Noise-resilient voltage and frequency synchronisation of an autonomous microgrid. *IET Generation, Transmission and Distribution*, 13(2), 189–200. <https://doi.org/10.1049/iet-gtd.2018.6409>
- Villarrubia, G.; J. F. Paz; D. H. De Iglesia; D. La and J. Bajo. (2017).** *Combining Multi-Agent Systems and Wireless Sensor Networks for Monitoring Crop Irrigation*. <https://doi.org/10.3390/s17081775>
- Wang, Q. I. and Wang, J. (2018).** Fully Distributed Fault-Tolerant Consensus Protocols for Lipschitz Nonlinear Multi-Agent Systems. *IEEE Access*, 6, 17313–17325. <https://doi.org/10.1109/ACCESS.2018.2821918>
- Xianji, J.; L. Jianquan; T. Weiming; L. Lei and L. Zhongwei. (2017).** *Multi-agent trust-based intrusion detection scheme for wireless sensor networks*. 0, 1–12. <https://doi.org/10.1016/j.compeleceng.2017.04.013>
- Yanmaz, E.; S. Yahyanejad; B. Rinner; H. Hellwagner and C. Bettstetter. (2018).** Drone networks: Communications, coordination, and sensing. *Ad Hoc Networks*, 68, 1–15. <https://doi.org/10.1016/j.adhoc.2017.09.001>
- Zhang, H.; S. Kim; Q. Sun and J. Zhou. (2017).** Distributed Adaptive Virtual Impedance Control for Accurate Reactive Power Sharing Based on Consensus Control in Microgrids. *IEEE Transactions on Smart Grid*, 8(4), 1749–1761. <https://doi.org/10.1109/TSG.2015.2506760>