IMPROVING PERFORMANCE FOR MULTI-AGENT SYSTEMS USING MIXED-TYPE FEEDBACK AND FUZZY LOGIC TUNING

by

Lucas Wan

Submitted in partial fulfillment of the requirements
for the degree of Master of Applied Science

at

Dalhousie University
Halifax, Nova Scotia
July 2021

© Copyright by Lucas Wan, 2021

Table of Contents

List of Tables . . . . **v**

List of Figures . . . . **vi**

Abstract . . . . **ix**

List of Abbreviations and Symbols Used . . . . **x**

Acknowledgements . . . . **xii**

Chapter 1 Introduction **1**

1.1 Applications of Multi-Agent Systems 1
1.1.1 Search and Rescue 1
1.1.2 Collaborative Manufacturing 2

1.2 Research Motivation 3

1.3 Thesis Contribution and Outline 4

Chapter 2 Literature Review **5**

2.1 Leader-Follower Communication 5

2.2 Communication Constraints 5

2.3 Sliding Mode Control 6

2.4 Fuzzy Logic in Control Systems 6

Chapter 3 Background Theory **8**

3.1 Communication Theory 8
3.1.1 Graph Theory 8
3.1.2 Time-Delay Model 10

3.2 Performance Metrics 10

3.3 Mathematical Theory 11
3.3.1 Kronecker Product 11
3.3.2 Schwarz's Inequality 11
3.3.3 Lyapunov Stability 12
3.3.4 Lipschitz Condition 12

3.4 Non-Singular Terminal Sliding Mode Control 13

3.5 Phantom Omni Model 14
3.5.1 Forward Kinematic Model 15
3.5.2 Dynamic Model 16

Chapter 4 Improving Performance for Linear MASs 19

4.1 Problem Formulation 19

4.2 Control Development 19
4.2.1 Controller Design 20
4.2.2 Error Dynamics 20
4.2.3 Stability Analysis 22
4.2.4 Fuzzy Logic Control 24

4.3 Simulations 27
4.3.1 Simulation Model 27
4.3.2 Gain Feasibility Results 29
4.3.3 Simulation Results 29

4.4 Summary 37

Chapter 5 Improving Performance for Euler-Lagrange MASs . . . 38

5.1 Problem Formulation 38

5.2 Control Development 39
5.2.1 Error Dynamics 39
5.2.2 Controller Design 41
5.2.3 Stability Analysis 42
5.2.4 Error Boundedness 44
5.2.5 Fuzzy Logic Control 45

5.3 Simulations and Experiments 47
5.3.1 Simulation Model 47
5.3.2 Simulation Results 49
5.3.3 Experimental Setup 59
5.3.4 Experimental Results 60

5.4 Summary 64

Chapter 6 Conclusions 65

6.1 Conclusions 65

6.2 Future Work 65

Bibliography **67**

Appendix A Author's Publications **72**

Appendix B Journal Paper Copyright Agreement **73**

List of Tables

3.1 Estimated parameters for Phantom Omni dynamic model . . . 15

4.1 Linguistic terms 26

4.2 Rule base for K_{p_f} 26

4.3 Rule base for K_{d_f} 27

4.4 Rule base for κ_{1_f} 27

4.5 Simulation performance metrics for Cases (a) and (b) 32

4.6 FLC parameters for Cases (a) and (b) 32

5.1 Rule base for $\Delta\boldsymbol{\kappa}_i$ and $\Delta\boldsymbol{\varpi}_{1_i}$ 47

5.2 FLC parameters for NTSMC+FLC simulations 55

5.3 Simulation performance metrics 58

List of Figures

1.1 Ground robot used for military applications [1] 2

1.2 Multiple manipulators used for manufacturing applications [2] 3

3.1 Example of a four agent leader-follower communication topology 9

3.2 Example of a time delay between agent i and j, $T_{ij}(t)$, where $\bar{T}_{ij} = 1.25ms$ and $\bar{\delta}_{ij} = 0.25ms$ 10

3.3 Phantom Omni device with link lengths and joint angle convention labels [3] 14

4.1 (a) Input membership functions and (b) output membership functions for PD+FLC 26

4.2 Leader-follower communication topology for linear simulations with (a) five agents (b) seven agents 28

4.3 Feasibility range for topology (a) with $\bar{T}_{ij_{max}} = 0.023\ s$ 29

4.4 Feasibility range for topology (a) with $\bar{T}_{ij_{max}} = 0.033\ s$ 29

4.5 Feasibility range for topology (b) with $\bar{T}_{ij_{max}} = 0.023\ s$ 29

4.6 Feasibility range for topology (b) with $\bar{T}_{ij_{max}} = 0.033\ s$ 29

4.7 RMSE with various constant values of κ_1 for Case (a) 30

4.8 Agent position and position error for $\kappa_1 = 1.00$ for Case (a) . . 30

4.9 Agent position and position error for $\kappa_1 = 0.75$ for Case (a) . . 31

4.10 Agent position and position error for $\kappa_1 = 0.50$ for Case (a) . . 31

4.11 Agent position and position error for FLC tuned κ_1 for Case (a) 33

4.12 Time-varying K_p, K_d and κ_1 tuned with FLC for Case (a) . . 33

4.13 RMSE with various constant values of κ_1 for Case (b) 34

4.14 Agent position and position error for $\kappa_1 = 1.00$ for Case (b) . 34

4.15 Agent position and position error for $\kappa_1 = 0.75$ for Case (b) . 35

4.16 Agent position and position error for $\kappa_1 = 0.50$ for Case (b) . 35

4.17 Agent position and position error for FLC tuned κ_1 for Case (b) 36

4.18 Time-varying K_p, K_d and κ_1 tuned with FLC for Case (b) . . 36

5.1 (a) Input membership functions and (b) output membership functions for NTSMC+FLC 47

5.2 Leader-follower communication topology for EL simulations . 48

5.3 The disturbances in task and joint space, $\boldsymbol{d}_2^u$ and $\boldsymbol{d}_2^\tau$, applied to Agent 2, for simulation with $\kappa = 30$ and $\varpi_1 = 1.00$ 49

5.4 Simulation RMSE with various constant values of $\boldsymbol{\kappa}$ and $\boldsymbol{\varpi}_1$ for the topology in Fig. 5.2 50

5.5 Simulation maximum error with various constant values of $\boldsymbol{\kappa}$ and $\boldsymbol{\varpi}_1$ for the topology in Fig. 5.2 50

5.6 Simulation RMSE with various constant values of $\boldsymbol{\kappa}$ and $\boldsymbol{\varpi}_1$ for a fully connected topology of five agents 51

5.7 Simulation maximum error with various constant values of $\boldsymbol{\kappa}$ and $\boldsymbol{\varpi}_1$ for a fully connected topology of five agents 51

5.8 Simulation RMSE with various constant values of $\boldsymbol{\kappa}$ and $\boldsymbol{\varpi}_1$ for a fully connected topology of three agents 52

5.9 Simulation maximum error with various constant values of $\boldsymbol{\kappa}$ and $\boldsymbol{\varpi}_1$ for a fully connected topology of three agents 52

5.10 Simulation agents' end effector positions for $\boldsymbol{\kappa} = 30$ and $\boldsymbol{\varpi}_1 = 1.00$ with NTSMC 53

5.11 Simulation agents' end effector positions for $\boldsymbol{\kappa} = 30$ and $\boldsymbol{\varpi}_1 = 0.95$ with NTSMC 54

5.12 Simulation agents' end effector positions for $\boldsymbol{\kappa} = 30$ and $\boldsymbol{\varpi}_1 = 0.90$ with NTSMC 55

5.13 Simulation agents' end effector positions for $\boldsymbol{\kappa} = 5$ and $\boldsymbol{\varpi}_1 = 1.00$ with NTSMC 56

5.14 Simulation agents' end effector positions for $\boldsymbol{\kappa} = 5$ and $\boldsymbol{\varpi}_1 = 0.80$ with NTSMC 56

5.15 Simulation agents' end effector positions for $\boldsymbol{\kappa}_i$ and $\boldsymbol{\varpi}_{1_i}$ with NTSMC+FLC 57

5.16 Simulation time-varying $\boldsymbol{\kappa}_i$ tuned with NTSMC+FLC 57

5.17 Simulation time-varying $\boldsymbol{\varpi}_{1_i}$ tuned with NTSMC+FLC 58

5.18 Experimental setup of two Phantom Omni devices 59

5.19 Leader-follower communication topology for multiple Phantom Omni systems 60

5.20 Experimental agents' end effector positions for Trial 1 61

5.21 Experimental time-varying $\boldsymbol{\kappa}_i$ for Trial 1 61

5.22 Experimental time-varying $\boldsymbol{\varpi}_{1_i}$ for Trial 1 62

5.23 Experimental agents' end effector positions for Trial 2 62

5.24 Experimental time-varying $\boldsymbol{\kappa}_i$ for Trial 2 63

5.25 Experimental time-varying $\boldsymbol{\varpi}_{1_i}$ for Trial 2 63

Abstract

The synchronization of multi-robot systems has numerous applications, including search and rescue missions and collaborative manufacturing. In multi-agent systems, information transmission delays in the communication network can lead to the degradation of overall system performance. The work in this thesis leverages the existence of delayed information in the controller feedback and proposes a method of tuning the controller parameters online with fuzzy logic to improve the synchronization performance. Teams of direct current motors and teams of Euler-Lagrange manipulators are considered. The stabilities of the proposed linear and nonlinear control policies are analyzed by defining mixed-type error dynamics and using linear matrix inequality techniques and Lyapunov's method. Simulations are conducted to observe the effect of introducing self-delayed states into the controller feedback for different networks and the effectiveness of the proposed fuzzy logic control method. Experimental testing of a group of Phantom Omni manipulators are performed to validate the proposed controller.

List of Abbreviations and Symbols Used

DC	direct current
DOF	degree-of-freedom
EL	Euler-Lagrange
FLC	fuzzy logic control
I	integral
ISTAR	intelligence, surveillance, target acquisition, and reconnaissance
LAN	local area network
LMI	linear matrix inequality
MAS	multi-agent system
NTSMC	non-singular terminal sliding mode control
PD	proportional-derivative
PI	proportional-integral
PID	proportional-integral-derivative
RMSE	root mean squared error
SMC	sliding mode control
UDP	user datagram protocol

$\sum$	summation
$\in$	belongs to
$\circ$	element-wise multiplication
$\otimes$	Kronecker product
∞	infinity
$A > 0$	positive definite matrix A
A^{-1}	inverse of matrix A
A^T	transpose of matrix A
I_n	$n \times n$ identity matrix
T_{ij}	time-varying delay from agent j to agent i

$\bar{T}_{ij}$	average delay from agent j to agent i
δ_{ij}	delay deviation from agent j to agent i
$\mathcal{A}$	adjacency matrix
$\mathcal{B}$	leader-connectivity matrix
$\mathcal{D}$	in-degree matrix
$\mathcal{E}$	edge set of a graph
$\mathcal{G}$	graph
$\mathcal{L}$	Laplacain matrix
$\mathcal{V}$	node set of a graph
$\mathbb{R}$	set of real numbers
$\mathbb{R}^n$	set of $n \times 1$ real vectors
$\mathbb{R}^{n\times n}$	set of $n \times n$ real matrices

Acknowledgements

I am grateful for my supervisor, Dr. Ya-Jun Pan, for her consistent support and guidance. I am fortunate to have had her exceptional advice and encouragement throughout the years. I would like to thank Dr. Ted Hubbard and Dr. Guy Kember for their input and assistance as my supervisory committee members. I would also like to thank my colleagues in the Advanced Control and Mechatronics Lab for their help and company throughout this degree. Finally, I would like to express my gratitude towards my friends and family for their support during the development of this thesis and in all other aspects of my life.

Chapter 1

Introduction

This chapter describes the importance of multi-agent systems (MASs) for practical applications in various industries and gives an overview of the motivation, contribution, and outline of this thesis.

1.1 Applications of Multi-Agent Systems

In the field of robotics, MASs have a broad scope of applications in numerous industries. Tasks suitable for robotic MASs include military intelligence, surveillance, target acquisition, and reconnaissance (ISTAR), military demining, altitude alignment of satellites, marine exploration and sampling, and autonomous transportation. Two potential applications discussed in this chapter are military search and rescue missions and smart collaborative manufacturing.

1.1.1 Search and Rescue

During disaster relief, after situations such as earthquakes, floods, wildfires, terrorist attacks, and nuclear explosions, search and rescue missions are employed to locate, rescue, and medically assist victims [4]. The deployment of an intelligent, cooperative team of robotic MASs in a search and rescue scenario can minimize injuries to rescue workers and victims, improve the response time, and increase the operating range to otherwise inaccessible areas [5]. A diverse team of multiple robots can provide extensive information on the environment from various locations and reduce operating costs by optimizing the distribution of sensors and equipment between different robots to minimize payload requirements [6]. Fig. 1.1 shows an example of a robot used for a search and rescue mission.

Figure 1.1: Ground robot used for military applications [1]

1.1.2 Collaborative Manufacturing

Cooperative robotic manipulators are becoming increasingly prevalent in the manufacturing industry as robots become progressively cost-effective, intelligent, and adaptable and Industry 4.0 continues to develop and popularize. Industry 4.0 offers increased automation and minimized human intervention through machine-to-machine communication and cloud-based data collection, and cooperative smart robotic manipulators play a major role in this revolution of traditional manufacturing. The use of multiple robotic arms can grant higher payload manipulation and allow for more complex manufacturing operations. In an assembly line, there is often a requirement for numerous manipulators to operate simultaneously on the same task or operate collaboratively on a common task in a safe and efficient manner [7]. Fig. 1.2 shows an example of multiple robots used for an industrial manufacturing process.

Figure 1.2: Multiple manipulators used for manufacturing applications [2]

1.2 Research Motivation

Compared to an individual robot, a team of robots can achieve higher operating efficiency, are more robust in response to hardware and software failures, and can complete complex tasks. To avoid a single point of failure, MAS controllers are designed to be decentralized, such that each agent is operating individually and only using local information of itself and its communicating neighbors.

The synchronization of a group of networked agents is an important control task where the objective is to direct the states of all agents to converge to a common value. In MASs, the terms synchronization and consensus can be interchangeable, though synchronization typically refers to converging to a time-varying trajectory, whereas consensus typically refers to converging to a constant value. The successful completion of MAS tasks is often evaluated by the accuracy of synchronizing the states of interest for all agents.

In most applications, agents in MASs communicate through a wireless network, which can have limitations such as delays in data transmission. The presence of delays in the network can reduce the synchronization tracking accuracy of a MAS and cause the system to become unstable. The motivation of this work is to minimize the adverse effects of network delays on the synchronization performance of MASs in the context of robotics.

1.3 Thesis Contribution and Outline

The primary contribution of this thesis is the development of novel control methods for linear and nonlinear MASs with time-varying communication network delays. The objective of the proposed controllers is to improve the synchronization tracking performance for teams of leader-follower MASs. The decentralized controllers are designed with mixed-type feedback, which considers each agent's error at the current time and at the estimated delayed time. Time-varying gains and proportions of the delayed feedback error are used to adapt the controller to various factors, such as external disturbances and friction. Fuzzy logic is employed as a nonlinear technique to tune the control parameters online. Lyapunov and linear matrix inequality (LMI) techniques are employed to prove the stability of the control policies. Extensive simulations are conducted to identify the impact of the proposed controllers on the overall system performance and demonstrate the importance of selecting suitable control parameters. Experiments are conducted with real-time devices to validate the control policy in the presence of network delays and external disturbances.

The contents of this thesis are organized as follows. Chapter 1 introduces the importance of robotic MASs in various industries and the significance of this work. Chapter 2 presents literature that has provided a foundation for this research. The literature is related to leader-follower topologies, MASs with limited communication, sliding mode control (SMC), and applications of fuzzy logic in conventional control problems. Chapter 3 provides important background in graph theory, kinematic and dynamic modeling, Lyapunov's method, and various mathematical theories that are required for the remaining chapters. Chapter 4 and Chapter 5 describe the controller design, stability proof, and results for linear MASs and Euler-Lagrange (EL) MASs, respectively. Chapter 6 concludes the preceding work and proposes several prospective areas for future research.

Chapter 2

Literature Review

This chapter presents recent literature on the topics of leader-follower communication topologies, MASs with network delays, applications of SMC, and the use of fuzzy logic control (FLC) in linear and nonlinear control. This chapter highlights the areas of research that have not been explored in existing literature that are addressed in this thesis.

2.1 Leader-Follower Communication

The communication networks employed in this work take the form of leader-follower topologies. The leader-follower control approach is common for the synchronization, consensus, and formation problems in MASs, where the leader does not receive information from the follower agents and acts independently [8]. It is required that at least one follower receives information directly from the leader and in many applications, the leader is a virtual agent [9, 10]. There have been numerous approaches in developing decentralized controllers for MAS consensus that adopt the leader-follower communication topology [11, 12, 13]. In [14], a decentralized controller is proposed for the synchronization of uncertain EL agents in an ideal leader-follower communication network. The work in [15] applies non-singular terminal sliding mode control (NTSMC) with constant parameters to synchronize a group of follower manipulators with a virtual leader.

2.2 Communication Constraints

Communication delays in the information exchange between agents, which can be a result of limited bandwidth, excessive data transmission, or actuator response time, can cause undesired oscillations and instability in MASs [16]. These delays can introduce a significant latency between the states of the agents, particularly if the follower agents are receiving information indirectly from the leader agent, and can decrease the tracking performance of the MAS [17]. Methods of ensuring stability of MASs with time-varying communication delays have been presented in [18, 19, 20]. With the presence of time-delay in the network there exists two independent types of feedback, feedback without self-delay and feedback with self-delay [21]. The combination of feedback with and without self-delay is termed as mixed-type feedback and can be integrated in the controller design, where the self-delay is approximated as the average of the known time-varying delay [22]. Work in [23] has proposed a mixed feedback controller for a group of EL systems. Another special form of mixed feedback controller was proposed in [22], which showed a promising leader-following results in a multi-manipulator system. However, the controllers in previous literature are applied with constant weightings of self-delayed feedback.

2.3 Sliding Mode Control

EL systems are used to model nonlinear systems, with practical applications in robotic manipulators, autonomous underwater vehicles, and legged robots [19]. Due to the complexity of nonlinear systems, it is difficult to determine the exact EL model, introducing uncertainties to the system. SMC can compensate for unknown external disturbances, friction, and uncertainties and is commonly used as it is robust and easy to design and implement [24]. Recent work has been done on implementing SMC for the consensus of multiple nonlinear systems [25, 26]. NTSMC is advantageous over the conventional SMC as it can converge in a finite time, results in higher performance, and avoids the singularity problem caused by the nonlinear switching surface [27]. In [28] and [29], NTSMC methods are applied to nonlinear MASs. In [30], a distributed sliding-mode control algorithm is proposed for a group of EL agents. These methods are limited to constant control gains and are applied to networks without delays.

2.4 Fuzzy Logic in Control Systems

Fuzzy logic provides a nonlinear mathematical representation of noisy, uncertain, and imprecise information. In control applications, FLC is advantageous over other nonlinear approaches in that it provides a representation based on human perception and therefore the user can easily integrate their experience into the controller design.

The performance of conventional proportional-integral-derivative (PID) controllers diminishes when time delays are introduced into the system [31], therefore adaptive PID controllers have been developed. A popular method is to use FLC for the online tuning of the PID gain parameters. There have been several works [32, 33], that have used FLC to tune PID controllers to improve the performance of individual direct current (DC) motors. While there has been significant research in applying FLC to a single system with state or control input time delays [34, 35], there has not been work that has applied FLC to linear MASs with time-varying delays.

FLC has also been used in literature as an adaptive control method for nonlinear systems. In [36], a FLC system is used to compensate for the uncertainties of a single Euler-Lagrange system. In [37], a proportional-derivative (PD)-based fuzzy sliding mode controller is used to improve the tracking performance of a 3-degree-of-freedom (DOF) upper-limb exoskeleton. Fuzzy-logic-based methods are commonly used as adaptive methods to approximate non-linearities of systems with time delays as in [38]. In [39], adaptive fuzzy logic systems are used to tune the parameters of a SMC policy to reduce the chattering of a single manipulator. However, limited work exists on integrating NTSMC and FLC for MASs with network constraints.

Chapter 3

Background Theory

This chapter introduces fundamental theories and concepts that are used throughout the remaining chapters of the thesis.

3.1 Communication Theory

This section presents the mathematical representation and preliminaries of multi-agent communication networks and the time-varying delay model used in this work.

3.1.1 Graph Theory

Graph theory can be used as a practical method of modeling the communication structure of MASs [40, 41, 42]. Once a network communication topology has been established, the system can be analyzed using sub-graphs and graph matrices.

A directed graph, also known as a digraph, is represented by $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathcal{A}\}$. The finite non-empty node set, $\mathcal{V} = \{1, 2, \ldots, N\}$, represents the agents. The edge set, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$, represents the communication channels between agents. The adjacency matrix $\mathcal{A} = [a_{ij}]_{N\times N} \in \mathbb{R}^{N\times N}$ describes the connectivity between agents i and j, where $a_{ij} = 1$ for $(j, i) \in \mathcal{E}$, otherwise $a_{ij} = 0$. A directed edge, $(i, j) \in \mathcal{E}$, indicates that the child node j can receive information from the parent node i. In the case of an undirected graph, the edges are bidirectional and $(i, j) \in \mathcal{E} \leftrightarrow (j, i) \in \mathcal{E}$.

The degree of an undirected graph, $d(n)$, is defined as the number of nodes adjacent to a given node, n. For a digraph, the set of neighbors of node i is defined as $\mathcal{N}_i = \{j \in \mathcal{V} | (j, i) \in \mathcal{E}\}$ and the in-degree of node i is defined as $d_i = \sum_{j=1}^{n} a_{ij}$. The degree of the digraph $\mathcal{G}$ is represented as $\mathcal{D} = \text{diag}\{d_1, d_2, \ldots, d_n\}$. The Laplacian matrix, $\mathcal{L}$, is given by $\mathcal{L} = \mathcal{D} - \mathcal{A}$. For undirected graphs, L is symmetric, whereas for directed graphs, L is generally asymmetric.

A directed path is a series of connected edges within a digraph. A digraph is considered strongly connected if each node has a directed path to every other node and considered complete if each node has an edge to every other node. A digraph is considered a directed tree if it has a single node that has directed paths to every other node and does not have a parent node. A directed spanning tree is a directed tree within a graph. The communication networks that are considered in this work are assumed to be directed trees with a single leader agent as the root node and are assumed strongly connected.

For a leader-follower network with a single leader, the leader agent can be defined as agent 0 and the connectivity between the leader and followers can be represented by the leader-connectivity matrix, $\mathcal{B} = [b_1 \; b_2 \; \dots \; b_N]^T$, where $b_i = 1$ if the i^{th} follower is connected to the leader, otherwise $b_i = 0$. In this case, the adjacency matrix, $\mathcal{A}_f$, represents the connectivity of the followers.

Fig. 3.1 shows an example of a simple leader-follower topology. The graph is described by the following general matrices

$$\mathcal{A} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \; \mathcal{D} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \; \mathcal{L} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ -1 & 2 & 0 & -1 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix}.$$

The graph in Fig. 3.1 can also be represented as

$$\mathcal{B} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}, \; \mathcal{A}_f = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}.$$

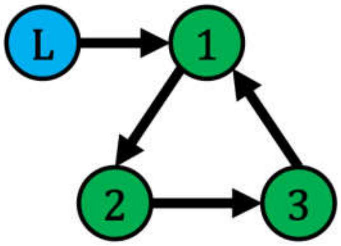

Figure 3.1: Example of a four agent leader-follower communication topology

3.1.2 Time-Delay Model

In this work, the information is assumed to be transmitted through user datagram protocol (UDP). UDP is a commonly used, low-latency network transport protocol [43].

The asynchronous time-varying delay between agent i and agent j can be represented as $T_{ij}(t) = \bar{T}_{ij} + \delta_{ij}(t)$, where $\bar{T}_{ij}$ is a constant average delay value and $\delta_{ij}(t)$ is the deviation, ranging from $-\bar{\delta}_{ij}$ to $\bar{\delta}_{ij}$. An example of $T_{ij}(t)$, measured through a wired LAN connection, is shown in Fig. 3.2.

Assumption 1. *The estimated communication delay, $\bar{T}$, is known. In practice, $\bar{T}$ may be approximated by identifying the network's communication protocols or by experimentally observing message timestamps between agents [23].*

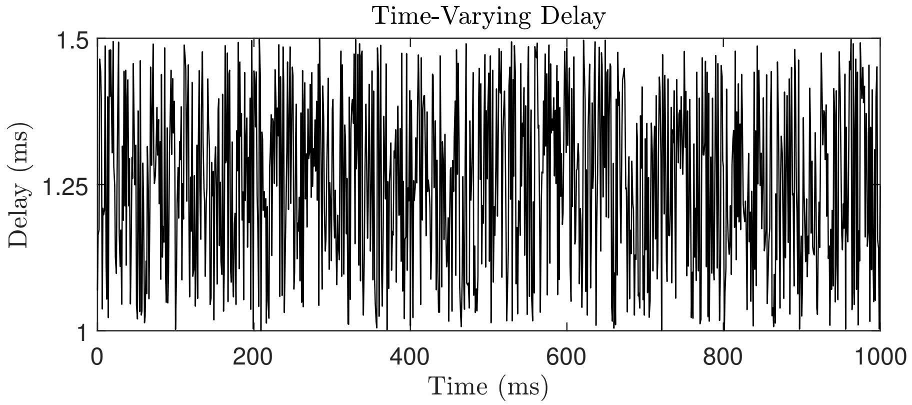

Figure 3.2: Example of a time delay between agent i and j, $T_{ij}(t)$, where $\bar{T}_{ij} = 1.25ms$ and $\bar{\delta}_{ij} = 0.25ms$

Remark 1. *The values for the average and deviation of the time-varying delays used in this work are chosen as the typical latency values in a wireless network for robots in an indoor environment [44]. Note that these network delays can be affected by numerous factors, such as the physical environment, the hardware of the agents, the relative motion of the agents, and the number of communicating agents in the network.*

3.2 Performance Metrics

In this work, the synchronization performance of a MAS is evaluated by two metrics. The first metric is the measure of the total root mean squared error (RMSE).

The RMSE evaluates the difference between leader's state, $\boldsymbol{x}_L(t)$, and each follower's states, $\boldsymbol{x}_i(t),\ i = 1, \ldots, \mathbb{N}$,

$$\text{RMSE} = \sum_{i=1}^{\mathbb{N}} \sum_{n=1}^{N} \sqrt{\frac{\sum_{r=1}^{P} \|\boldsymbol{x}_{L,n}(r) - \boldsymbol{x}_{i,n}(r)\|^2}{P}}, \tag{3.1}$$

where N is the number of states and P is the total number of sampling points.

The second performance metric is the maximum error, E_{max}, which is calculated as

$$E_{max} = \max_{i \in \mathbb{N}, n \in N, r \in P} \left(|\boldsymbol{x}_{L,n}(r) - \boldsymbol{x}_{i,n}(r)| \right). \tag{3.2}$$

3.3 Mathematical Theory

This section introduces the definitions and backgrounds of the fundamental mathematical theories used in the remaining chapters of the thesis.

3.3.1 Kronecker Product

The Kronecker product [45], represented by the symbol $\otimes$, of $\mathbf{A} \in \mathbb{R}^{\mathbf{m} \times \mathbf{n}}$ and $\mathbf{B} \in \mathbb{R}^{\mathbf{p} \times \mathbf{q}}$ results in matrix $\mathbf{C} \in \mathbb{R}^{(\mathbf{mp}) \times (\mathbf{nq})}$, defined as

$$\mathbf{C} = \mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} a_{11}\mathbf{B} & \ldots & a_{1n}\mathbf{B} \\ \vdots & \ddots & \vdots \\ a_{m1}\mathbf{B} & \ldots & a_{mn}\mathbf{B} \end{bmatrix}. \tag{3.3}$$

3.3.2 Schwarz's Inequality

For any two real integrable functions,$f(x)$ and $g(x)$, Shwarz's inequality [46] is defined as

$$\left[\int_{\mathbb{R}^n} f(x)g(x)dx \right]^2 \leq \int_{\mathbb{R}^n} [f(x)]^2 \, dx \int_{\mathbb{R}^n} [g(x)]^2 \, dx, \tag{3.4}$$

which holds equality if and only if $f(x) = \alpha g(x)$, where α is a constant.

3.3.3 Lyapunov Stability

The stability of a nonlinear system can proven by defining a Lyapunov function and ensuring that the function satisfies several conditions. The concept of Lyapunov theory is that the constructed function describes the energy of the physical system and the conditions ensure that the energy of the system decreases, guaranteeing stability [47]. The basis of Lyapunov theory can be described as follows. Consider an autonomous nonlinear system, $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$. The system is considered asymptotically stable at $\mathbf{x} = \mathbf{0}$ if there exists a Lyapunov function, $\mathbf{V}(\mathbf{x})$, that satisfies the following conditions [48]:

- The Lyapunov function is differentiable, continuous, and the initial conditions are zero: $\mathbf{V}(\mathbf{0}) = \mathbf{0}$.
- The Lyapunov function is positive definite: $\mathbf{V}(\mathbf{x}) > \mathbf{0}$.
- The derivative of the Lyapunov function is negative definite: $\dot{\mathbf{V}}(x) < 0$.

The system is considered globally asymptotically stable at $\mathbf{x} = \mathbf{0}$ if the following condition is satisfied [48]:

- The Lyapunov function goes to infinity as $\mathbf{x}$ goes to infinity: $\|\mathbf{x}\| \to \infty \Rightarrow \mathbf{V}(\mathbf{x}) \to \infty$.

Nonlinear controller design using Lyapunov theory involves defining a function of the system states and constructing a control policy that ensures that this function decreases [47]. A Lemma is introduced in Chapter 5, based on the work in [49], that proves the local finite-time stability and defines a bound for the settling time of a non-Lipschitz continuous nonlinear system using the Lyapunov stability theorem.

3.3.4 Lipschitz Condition

A function, $f(x)$, is considered Lipschitz if the condition

$$|f(a) - f(b)| \leq K|a - b|, \tag{3.5}$$

holds for some value of K, where $K \geq 0$ [50]. Therefore, a function is considered non-Lipschitz if no value of K satisfies (3.5), indicating that the derivative of a non-Lipschitz function is not bounded.

3.4 Non-Singular Terminal Sliding Mode Control

SMC is a robust nonlinear control method that is used for its ability to operate in the presence of uncertainties and external disturbances [51]. A NTSMC method is proposed in [52] to avoid the singularities that can occur in the traditional SMC. The basic NTSMC policy is constructed as follows. Consider a nonlinear system

$$\begin{cases} \dot{x}_1 &= x_2, \\ \dot{x}_2 &= f(\boldsymbol{x}) + g(\boldsymbol{x}) + b(\boldsymbol{x})u, \end{cases} \tag{3.6}$$

where $\boldsymbol{x} = [x_1, x_2]^T$ is the state vector, $f(\boldsymbol{x})$ and $b(\boldsymbol{x})$ are smooth nonlinear functions, $g(\boldsymbol{x})$ represents the external bounded disturbances, and u is the control input. A sliding variable is designed as

$$s = x_1 + \beta x_2^{\alpha}, \tag{3.7}$$

where $\beta > 0$ is a design constant and $\alpha = p/q$ with $p > 0$, $q > 0$ and p, q are adjacent odd numbers such that $1 < \alpha < 2$. A control law can be designed as a function of the system dynamics and a switching function. The switching function used in this work is sgn(s) which denotes the sign function, defined as

$$\operatorname{sgn}(s) = \begin{cases} 1 & \text{if } s > 0 \\ 0 & \text{if } s = 0 \\ -1 & \text{if } s < 0. \end{cases} \tag{3.8}$$

The control law is then commonly designed as

$$u = -b^{-1}(\boldsymbol{x})(f(\boldsymbol{x}) + \frac{\beta}{\alpha} x_2^{2-\alpha} + (G + \eta)\operatorname{sgn}(s)), \tag{3.9}$$

where G is the upper bound of the disturbances and η is a constant design parameter. The control input, u, is designed such that the sliding surface, s, can be reached in finite time by using Lyapunov based stability analysis method. In comparison to terminal sliding mode control, the NTSMC control law in (3.9) does not contain a term with x_1 in the denominator, which eliminates the risk of division by zero and encountering a singularity during the reaching phase or due to computational errors or uncertainties [52].

3.5 Phantom Omni Model

In this thesis, Phantom Omni haptic devices are used in the numerical and experimental case studies of EL MASs. The Phantom Omni is a device used for operators to touch and manipulate virtual objects. The device can provide haptic feedback, and therefore it can be used as a small-scale manipulator that is controlled by joint torque inputs. This section describes the forward kinematics and dynamic model used in simulations and experiments. The inverse dynamics are not considered in this work as it is assumed that the joint variables can be directly obtained from the devices. The Phantom Omni device is treated as a 3-DOF manipulator and is comprised of a single serial chain that links three actuated revolute joints. Here, the position of the end effector is represented by $\boldsymbol{x} \in \mathbb{R}^3$ and the joint angles are represented by $\boldsymbol{q} \in \mathbb{R}^3$. The diagram of the Phantom Omni device is shown in Fig. 3.3, outlining the reference frame used in the following sections and the estimated parameters are presented in Table 3.1.

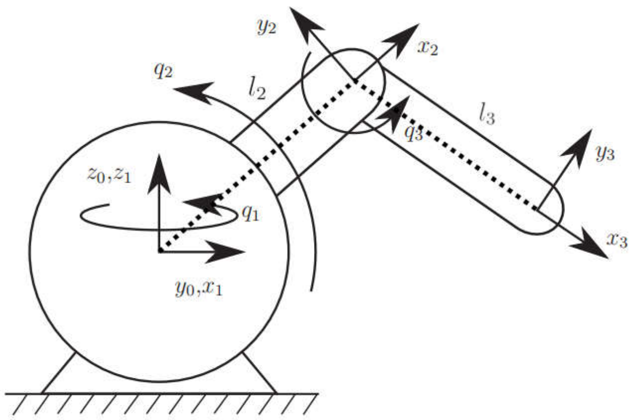

Figure 3.3: Phantom Omni device with link lengths and joint angle convention labels [3]

Table 3.1: Estimated parameters for Phantom Omni dynamic model

Parameter	Value	Parameter	Value
$g(kg \cdot m/s^2)$	9.8	$\phi_2(kg \cdot m^2)$	7.0×10^{-3}
$l_1(m)$	0	$\phi_3(kg \cdot m^2)$	8.0×10^{-3}
$l_2(m)$	0.130	$\phi_4(kg \cdot m^2)$	0.4×10^{-3}
$l_3(m)$	0.135	$\phi_5(kg \cdot m)$	9.1×10^{-3}
$\phi_1(kg \cdot m^2)$	3.7×10^{-3}	$\phi_6(kg \cdot m)$	5.2×10^{-3}

3.5.1 Forward Kinematic Model

The position, velocity, and acceleration of the end effector can be determined by the following functions of the joint angles, joint angular velocities, and joint angular accelerations:

$$\boldsymbol{x} = \begin{bmatrix} r\mathrm{c}_1 \\ r\mathrm{s}_1 \\ h \end{bmatrix}, \tag{3.10}$$

$$\dot{\boldsymbol{x}} = J\dot{\boldsymbol{q}}, \tag{3.11}$$

$$\ddot{\boldsymbol{x}} = J\ddot{\boldsymbol{q}} + \dot{J}\dot{\boldsymbol{q}}. \tag{3.12}$$

where $\mathrm{c}_i = \cos(\boldsymbol{q}(i))$, $\mathrm{s}_i = \sin(\boldsymbol{q}(i))$, $r = l_2\mathrm{c}_2 + l_3\mathrm{c}_{23}$, and $h = l_2\mathrm{s}_2 + l_3\mathrm{s}_{23}$. The link lengths, l_i, are given in Table 3.1. The Jacobian matrix and deriviative of the Jacobian matrix are defined as

$$J = \begin{bmatrix} -r\mathrm{s}_1 & h\mathrm{c}_1 & l_3\mathrm{c}_1\mathrm{s}_{23} \\ r\mathrm{c}_1 & -h\mathrm{s}_1 & -l_3\mathrm{s}_1\mathrm{s}_{23} \\ 0 & r & l_3\mathrm{c}_{23} \end{bmatrix},$$

$$\dot{J} = \begin{bmatrix} \dot{\boldsymbol{q}}^{\boldsymbol{T}} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & -\dot{\boldsymbol{q}}^{\boldsymbol{T}} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & -\dot{\boldsymbol{q}}^{\boldsymbol{T}} \end{bmatrix} \begin{bmatrix} -r\mathrm{c}_1 & h\mathrm{s}_1 & l_3\mathrm{s}_1\mathrm{s}_{23} \\ h\mathrm{s}_1 & -r\mathrm{c}_1 & -l_3\mathrm{c}_1\mathrm{c}_{23} \\ l_3\mathrm{s}_1\mathrm{s}_{23} & -l_3\mathrm{c}_1\mathrm{c}_{23} & -l_3\mathrm{c}_1\mathrm{c}_{23} \\ r\mathrm{s}_1 & h\mathrm{c}_1 & l_3\mathrm{c}_1\mathrm{s}_{23} \\ h\mathrm{c}_1 & r\mathrm{s}_1 & l_3\mathrm{s}_1\mathrm{c}_{23} \\ l_3\mathrm{c}_1\mathrm{s}_{23} & l_3\mathrm{s}_1\mathrm{c}_{23} & l_3\mathrm{s}_1\mathrm{c}_{23} \\ 0 & 0 & 0 \\ 0 & h & l_3\mathrm{s}_{23} \\ 0 & l_3\mathrm{s}_{23} & l_3\mathrm{s}_{23} \end{bmatrix},$$

where $\mathrm{c}_{ij} = \cos(\boldsymbol{q}(i) + \boldsymbol{q}(j))$ and $\mathrm{s}_{ij} = \sin(\boldsymbol{q}(i) + \boldsymbol{q}(j))$.

Remark 2. *When operating the Phantom Omni device in Simulink, the measured joint angles, $\boldsymbol{q}^m$, must be converted to match the convention used in the kinematic model, $\boldsymbol{q}$, as*

$$\boldsymbol{q} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -1 & 1 \end{bmatrix} \boldsymbol{q}^m - \begin{bmatrix} 0 \\ 0 \\ \frac{\pi}{2} \end{bmatrix}. \tag{3.13}$$

3.5.2 Dynamic Model

The dynamic model of the Phantom Omni device can be represented by an EL system as

$$M_i(\boldsymbol{q}_i)\ddot{\boldsymbol{x}}_i + C_i(\boldsymbol{q}_i, \dot{\boldsymbol{q}}_i)\dot{\boldsymbol{x}}_i + \boldsymbol{g}_i(\boldsymbol{q}_i) = \boldsymbol{u}_i + \boldsymbol{f}_i, \quad i \in \mathbb{N} \tag{3.14}$$

where $\boldsymbol{q}_i \in \mathbb{R}^k$ is the joint position vector, $\boldsymbol{x}_i \in \mathbb{R}^3$ is the end effector's position vector in the task space, $M_i \in \mathbb{R}^{3\times3}$ is the inertia matrix, $C_i \in \mathbb{R}^{3\times3}$ is the Coriolis and centripetal torque matrix, $\boldsymbol{g}_i \in \mathbb{R}^3$ is the gravitational torque vector, $\boldsymbol{u}_i \in \mathbb{R}^3$ is the control input, and $\boldsymbol{f}_i \in \mathbb{R}^3$ is the disturbance vector. The disturbance vector combines external disturbances and frictional forces as $\boldsymbol{f}_i = \boldsymbol{d}_i + \boldsymbol{f}_i^f(\dot{\boldsymbol{q}}_i)$, where $\boldsymbol{d}_i \in \mathbb{R}^3$ is the upper bounded external disturbances and $\boldsymbol{f}_i^f \in \mathbb{R}^3$ is the upper bounded frictional forces.

The parametric matrices and vector, M, C, and $\boldsymbol{g}$, of the Phantom Omni device in (3.14) are modeled as

$$M = \begin{bmatrix} h_{11} & 0 & 0 \\ 0 & h_{22} & h_{23} \\ 0 & h_{32} & h_{33} \end{bmatrix}, \boldsymbol{g} = \begin{bmatrix} 0 \\ g(\phi_5 c_2 + \phi_6 c_{23}) \\ g\phi_6 c_{23} \end{bmatrix},$$

$$C = \begin{bmatrix} -a_1\dot{\boldsymbol{q}}(1) + a_2\dot{\boldsymbol{q}}(3) & -a_1\dot{\boldsymbol{q}}(1) & -a_2\dot{\boldsymbol{q}}(1) \\ a_1\dot{\boldsymbol{q}}(1) & -a_3\dot{\boldsymbol{q}}(3) & -a_3(\dot{\boldsymbol{q}}(2) + \dot{\boldsymbol{q}}(3)) \\ a_2\dot{\boldsymbol{q}}(1) & a_3\dot{\boldsymbol{q}}(2) & 0 \end{bmatrix},$$

where

$$\begin{aligned} h_{11} &= \phi_1 + \phi_2 c_2^2 + \phi_3 c_{23}^2 + 2\phi_4 c_2, \\ h_{22} &= \phi_2 + \phi_3 + 2\phi_4 c_3, \\ h_{23} &= h_{32} = \phi_3 + \phi_4 c_3, \\ h_{33} &= \phi_3, \\ a_1 &= \phi_2 c_2 s_2 + \phi_3 c_{23} s_{23} + \phi_4 c_2 \sin(2\boldsymbol{q}(2) + \boldsymbol{q}(3)), \\ a_2 &= \phi_3 c_{23} s_{23} + \phi_4 c_2 s_{23}, \\ a_3 &= \phi_4 s_3. \end{aligned}$$

The values used to estimate the dynamics of the Phantom Omni device are given in Table 3.1.

In this work, the internal viscous frictional forces are considered negligible compared to the internal coulombic frictional forces [53]. The coulombic friction applied to the manipulators in simulation are modeled as

$$\boldsymbol{f}_i^f = \boldsymbol{f}_c \mathrm{sgn}(\dot{\boldsymbol{q}})\ Nm$$

where $\boldsymbol{f}_c = [0.006, 0.012, 0.010]^T\ Nm$ is the coulombic friction matrix, obtained experimentally in [54].

As a safety measure to avoid damaging the manipulators, the control input, $\boldsymbol{u}_i$, is limited to a saturated control input vector, $\boldsymbol{\theta}_i(\boldsymbol{u}_i) \in \mathbb{R}^k$, which is given as

$$\theta_{i,l}(u_{i,l}) = \begin{cases} \bar{u}_{i,l}\text{sgn}(u_{i,l}) & |u_{i,l}| \geq \bar{u}_{i,l}, \\ u_{i,l} & |u_{i,l}| < \bar{u}_{i,l}, \end{cases} \quad l = 1, \ldots, k \tag{3.15}$$

where $\bar{u}_{i,l}$ is the saturation bound, given by the hardware limitations of the agent, defined for the Phantom Omni devices as

$$\bar{\boldsymbol{u}}_i = \begin{bmatrix} 0.28 & 0.27 & 0.19 \end{bmatrix}^T Nm. \tag{3.16}$$

Chapter 4

Improving Performance for Linear MASs

This chapter introduces the framework of synchronizing a networked group of linear agents in the presence of asynchronous time-varying delays. The proposed control policy is presented and a method of deriving a stable range of gains from LMI conditions and Lyapunov criteria is established. An approach to tuning the control parameters online using FLC is introduced. Simulations of a group of linear DC motors are conducted to demonstrate an improvement in the synchronization tracking performance by implementing the proposed controller and tuning the control parameters online.

4.1 Problem Formulation

Consider a MAS composed of $\mathbb{N}$ homogeneous linear agents, represented by the following dynamics

$$\begin{aligned} \dot{\mathbf{x}}_i &= A\mathbf{x}_i + B\mathbf{u}_i, \\ \mathbf{y}_i &= C\mathbf{x}_i, \quad i \in \mathbb{N} \end{aligned} \tag{4.1}$$

where $\mathbf{x}_i \in \mathbb{R}^n$ is the state vector, $\mathbf{y}_i \in \mathbb{R}^p$ is the system output, $\mathbf{u}_i \in \mathbb{R}^m$ is the control input for the ith follower. The known system dynamics are described by $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, and $C \in \mathbb{R}^{p\times n}$.

Assumption 2. *In this chapter, the communication networks are assumed to be strongly connected. It is also assumed that the leader agent does not receive any information from the follower agents and that the communication between the followers is undirected.*

4.2 Control Development

In this section, a mixed-type feedback controller is proposed, the closed-loop error dynamics are derived, an LMI condition is established to compute a stable range of gains, and an FLC system is developed to tune the controller parameters as a function of the total error of the MAS.

4.2.1 Controller Design

With the existence of communication delays in the network, the delayed states of the agents can be introduced into the system, termed as the mixed-type error feedback, to help improve the synchronization performance [23, 22]. Mixed-type feedback is integrated into the control method developed in [55]. In this thesis, the resulting controller is proposed as

$$\begin{aligned}
\mathbf{u}_i &= \mathbf{u}_{i_a} + \mathbf{u}_{i_b}, \\
\mathbf{u}_{i_a} &= \mathbf{l}\mathbf{x}_i, \\
\mathbf{u}_{i_b} &= -\mathbf{K}\big(\frac{\kappa_1}{\mathcal{N}_i}\sum_{j\in\mathcal{N}_i} a_{ij}(\mathbf{x}_i(t) - \mathbf{x}_j(t-(\bar{T}_{ij}+\delta_{ij})) \\
&\quad + \frac{\kappa_2}{\mathcal{N}_i}\sum_{j\in\mathcal{N}_i} a_{ij}(\mathbf{x}_i(t-\bar{T}_{ij}) - \mathbf{x}_j(t-(\bar{T}_{ij}+\delta_{ij})))\big),
\end{aligned} \tag{4.2}$$

where $\mathbf{u}_{i_a}$ is the local feedback controller, $\mathbf{u}_{i_b}$ is the mixed-type feedback network interaction controller, a_{ij} are elements of the network adjacency matrix, and $\mathcal{N}_i$ is the neighbor set, used to compensate for scaling networks. The positive constant parameters, κ_1 and κ_2 are related by $\kappa_1 + \kappa_2 = 1$, where κ_1 defines the proportion of feedback without self-delay and κ_2 defines the proportion of feedback with self-delay. The local gain vector, $\mathbf{l}$, can be designed using the pole placement method and the network gain vector, $\mathbf{K}$, is designed using LMI and Lyapunov-based methods.

4.2.2 Error Dynamics

The closed-loop dynamics of (4.1) with the proposed controller (4.2) can be represented as

$$\begin{aligned}
\dot{\mathbf{x}}_i =& (A + B\mathbf{l})\mathbf{x}_i - C_1\sum_{j\in\mathcal{N}_i} a_{ij}(\mathbf{x}_i(t) - \mathbf{x}_j(t-(\bar{T}_{ij}+\delta_{ij}))) \\
& - C_2\sum_{j\in\mathcal{N}_i} a_{ij}(\mathbf{x}_i(t-\bar{T}_{ij}) - \mathbf{x}_j(t-(\bar{T}_{ij}+\delta_{ij}))),
\end{aligned} \tag{4.3}$$

where

$$C_1 = \frac{B\mathbf{K}\kappa_1}{\mathcal{N}_i},$$
$$C_2 = \frac{B\mathbf{K}\kappa_2}{\mathcal{N}_i}.$$

The system error for the ith agent is defined as $\mathbf{e}_i(t) = \mathbf{x}_i(t) - \mathbf{x}_1(t), i \in 2, \ldots, \mathbb{N}$. Subsequently, the error dynamics can be written as

$$\begin{aligned}\dot{\mathbf{e}}_i =&(A + B\mathbf{l})\mathbf{e}_i - C_1 \sum_{j\in\mathcal{N}_i} a_{ij}(\mathbf{e}_i(t-\bar{T}_{ij}) - \mathbf{e}_j(t-\bar{T}_{ij})) - C_1 \sum_{j\in\mathcal{N}_i} a_{ij}\hat{\mathbf{e}}_{ij} - C_1 \sum_{j\in\mathcal{N}_i} a_{ij}\tilde{\mathbf{e}}_i \\ &- C_1 \sum_{j\in\mathcal{N}_1} a_{1j}\mathbf{e}_j(t-\bar{T}_{ij}) + C_1 \sum_{j\in\mathcal{N}_1} a_{1j}\hat{\mathbf{e}}_{1j} - C_2 \sum_{j\in\mathcal{N}_i} a_{ij}(\mathbf{e}_i(t-\bar{T}_{ij}) - \mathbf{e}_j(t-\bar{T}_{ij})) \\ &- C_2 \sum_{j\in\mathcal{N}_i} a_{ij}\hat{\mathbf{e}}_{ij} - C_2 \sum_{j\in\mathcal{N}_1} a_{1j}\mathbf{e}_j(t-\bar{T}_{ij}) + C_2 \sum_{j\in\mathcal{N}_1} a_{1j}\hat{\mathbf{e}}_{1j}, \qquad (4.4)\end{aligned}$$

where

$$\tilde{\mathbf{e}}_i = \mathbf{x}_i(t) - \mathbf{x}_i(t-\bar{T}_{ij}), \qquad (4.5)$$
$$\hat{\mathbf{e}}_{ij} = \mathbf{x}_j(t-\bar{T}_{ij}) - \mathbf{x}_j(t-(\bar{T}_{ij}+\delta_{ij})). \qquad (4.6)$$

The overall system error dynamics can be written as

$$\dot{\mathbf{e}} = \bar{A}\mathbf{e} + \bar{L}\mathbf{e}(t-\bar{T}_{ij}) + \Gamma\Delta, \qquad (4.7)$$

where

$$\begin{aligned}\bar{A} &= \mathbf{I}_{N-1} \otimes (A + B\mathbf{l}), \\ \Delta &= \begin{bmatrix}\hat{\mathbf{e}} & \tilde{\mathbf{e}}\end{bmatrix}^T, \\ \Gamma &= \begin{bmatrix}\hat{L} & \tilde{L}\end{bmatrix}, \\ \mathbf{e} &= \begin{bmatrix}\mathbf{e}_2^T & \mathbf{e}_3^T & \cdots & \mathbf{e}_N^T\end{bmatrix}^T, \\ \tilde{\mathbf{e}} &= \begin{bmatrix}\tilde{\mathbf{e}}_2^T & \tilde{\mathbf{e}}_3^T & \cdots & \tilde{\mathbf{e}}_N^T\end{bmatrix}^T, \\ \hat{\mathbf{e}} &= \begin{bmatrix}\hat{\mathbf{e}}_{11}^T & \hat{\mathbf{e}}_{12}^T & \cdots & \hat{\mathbf{e}}_{1N}^T & \cdots & \hat{\mathbf{e}}_{N1}^T & \hat{\mathbf{e}}_{N2}^T & \cdots & \hat{\mathbf{e}}_{NN}^T\end{bmatrix}^T,\end{aligned}$$

$$
\begin{aligned}
\bar{L} &= L_1 \otimes C_3, \\
\tilde{L} &= L_2 \otimes C_1, \\
\hat{L} &= L_3 \otimes -C_3, \\
L_1 &= \begin{bmatrix}
-a_{12} - \sum_{j=1}^{N} a_{2j} & -a_{13} + a_{23} & \cdots & -a_{1N} + a_{2N} \\
-a_{12} + a_{32} & -a_{13} - \sum_{j=1}^{N} a_{3j} & \cdots & -a_{1N} + a_{3N} \\
\vdots & \vdots & \ddots & \vdots \\
-a_{12} + a_{N2} & \cdots & \cdots & -a_{1N} - \sum_{j=1}^{N} a_{Nj}
\end{bmatrix}, \\
L_2 &= \begin{bmatrix}
-\sum_{j=1}^{N} a_{2j} & 0 & \cdots & 0 \\
0 & -\sum_{j=1}^{N} a_{3j} & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & \cdots & \cdots & -\sum_{j=1}^{N} a_{Nj}
\end{bmatrix}, \\
L_3 &= \begin{bmatrix}
a_{11} & \cdots & a_{1N} & -a_{21} & \cdots & -a_{2N} & 0 & \cdots & 0 & \cdots & \cdots & 0 & \cdots & 0 \\
a_{11} & \cdots & a_{1N} & 0 & \cdots & 0 & -a_{31} & \cdots & -a_{3N} & \cdots & \cdots & 0 & \cdots & 0 \\
 & \vdots & & & \vdots & & & \vdots & & \cdots & \cdots & & \vdots & \\
a_{11} & \cdots & a_{1N} & 0 & \cdots & 0 & 0 & \cdots & 0 & \cdots & \cdots & -a_{N1} & \cdots & a_{NN}
\end{bmatrix}, \\
C_3 &= C_1 + C_2 = \frac{B\mathbf{K}}{\mathcal{N}_i}.
\end{aligned}
$$

Remark 3. *Since the communication topology follows a leader-follower structure, $a_{1j} = 0, i \in 2, \ldots, \mathbb{N}$ and because the communication networks considered in this work are bidirectional, the matrices, $L_i, i = \{1, 2, 3\}$, are symmetric.*

The linear system in (4.1) with the proposed controller (4.2) will reach consensus and synchronize if the overall system error dynamics (4.7) is stable.

4.2.3 Stability Analysis

Theorem 1. *With the existence of asynchronous time-varying delays δ_{ij}, the system (4.1) achieves consensus asymptotically under the control law (4.2) for a given control* *gain vector $\boldsymbol{K} \in \mathbb{R}^n$ and positive scalars γ and $\bar{T}_{ij}$, if there exists symmetric positive definite matrices P_{11}, P_{22}, Q_{11}, Q_{22}, R_{11}, R_{22}, $S \in \mathbb{R}^{n\times n}$, and arbitrary matrices N_f, $M_f \in \mathbb{R}^{n\times n}$, $f \in 1, \cdots, 5$, that satisfy*

$$H_i = \begin{bmatrix} H_{i_{11}} + S & * & * & * & * & * \\ H_{i_{21}} & H_{i_{22}} & * & * & * & * \\ H_{i_{31}} & H_{i_{32}} & H_{i_{33}} & * & * & * \\ H_{i_{41}} & H_{i_{42}} & H_{i_{43}} & H_{i_{44}} & * & * \\ H_{i_{51}} & H_{i_{52}} & H_{i_{53}} & H_{i_{54}} & H_{i_{55}} & * \\ M_1^T & M_2^T & M_3^T & M_4^T & M_5^T & -\frac{1}{\gamma}\boldsymbol{I}_n \end{bmatrix} < 0, i \in 1, \cdots, \mathbb{N}-1, \tag{4.8}$$

where

$$\begin{aligned}
H_{i_{11}} &= (N-2)Q_{11} + \bar{T}_{ij}(N-2)R_{11} + N_1 + N_1^T + M_1(A + B\boldsymbol{l}) + (A + B\boldsymbol{l})^T M_1^T, \\
H_{i_{21}} &= -N_1^T + N_2 + M_2(A + B\boldsymbol{l}) + \lambda_i C_3^T M_1^T, \\
H_{i_{22}} &= -(N-2)Q_{11} - N_2 - N_2^T + \lambda_i M_2 C_3 + \lambda_i C_3^T M_2^T, \\
H_{i_{31}} &= (N-2)P_{11} + N_3 - M_1 + M_3(A + B\boldsymbol{l}), \\
H_{i_{32}} &= -N_3 + \lambda_i M_3 C_3 - M_2^T, \\
H_{i_{33}} &= (N-2)Q_{22} + \bar{T}_{ij}(N-2)R_{22} - M_3 - M_3^T, \\
H_{i_{41}} &= (N-2)P_{22} + N_4 + M_4(A + B\boldsymbol{l}), \\
H_{i_{42}} &= -(N-2)P_{22} - N_4 + \lambda_i M_4 C_3, \\
H_{i_{43}} &= -M_4, \\
H_{i_{44}} &= -\frac{N-2}{\bar{T}_{ij}} R_{11}, \\
H_{i_{51}} &= N_5 - N_1^T + M_5(A + B\boldsymbol{l}), \\
H_{i_{52}} &= -N_5 - N_2^T + \lambda_i M_5 C_3, \\
H_{i_{53}} &= -N_3^T - M_5, \\
H_{i_{54}} &= -N_4^T, \\
H_{i_{55}} &= -N_5 - \frac{N-2}{\bar{T}_{ij}} R_{22} - N_5^T,
\end{aligned}$$

and λ_i is the ith eigenvalue of L_1 arranged in ascending order. The consensus error is bounded by

$$||\boldsymbol{e}(t)|| \leq \frac{\gamma^{-1}\lambda_{max}(\Gamma^T\Gamma)||\Delta||^2}{\lambda_{min}(S)}, \tag{4.9}$$

where $\lambda_{max}(\cdot)$ and $\lambda_{min}(\cdot)$ denote the maximum and minimum eigenvalues.

Remark 4. *The bounded consensus error defined in (4.9) is a function of Γ, Δ, γ and S, where Γ is dependent on the adjacency matrix, $\mathcal{A}$, and the gain vector, $\boldsymbol{K}$, Δ is dependent on the self-delay error of each agent (4.5), and γ and S are design parameters. The design of γ and S offer a trade-off between reducing the error bound, $||\boldsymbol{e}(t)||$, and reducing the feasibility of the inequality (4.8) [55].*

The inequality (4.8) is similar as presented in [55], where the Lyapunov function and the corresponding stability proof can be found. Compared to the work in [55], rather then computing the static gain vector, **K**, the inequality (4.8) is constructed and run iteratively to derive a feasible, stable range of gain values that the controller can operate within.

4.2.4 Fuzzy Logic Control

In general, the FLC tuning is applicable for the controller gain **K** design. FLC was chosen for the online tuning method of the controller parameters because it is easy to implement and update, it is suited to tolerate the randomness and non-linearities of the time-varying delays, and it can be developed based on user experience and the response of the system. For a simple case of a system of agents with typical second order dynamics, a PD type controller is used as a thorough demonstration in this chapter to demonstrate the tuning method. The PD controller is chosen for several reasons. The first is that tracking both the position and velocity are important in the synchronization scenario. The second is to avoid the wind-up error associated with the integral (I) term and the third is that tuning a PD controller is typically more straightforward then a proportional-integral (PI) or PID controller. The PD gain matrix is thus defined as follows with appropriate dimensions

$$\mathbf{K} = \begin{bmatrix} K_p, & K_d \end{bmatrix}. \tag{4.10}$$

The results in [23] show that increasing the self-delay in the feedback requires lower gains to maintain stability and performance. FLC is used to tune the gains and proportion of self-delay in the feedback based on the system error, acting as an adaptive controller. This presents the opportunity to increase the proportion of self-delay as the gains are lowered, improving the performance of the system. The FLC-based PD controller receives the total error, $\mathbf{s}$, as an input which is defined as

$$\mathbf{s} = \begin{bmatrix} s_p, & s_v \end{bmatrix},$$

where s_p is the position error and s_v is the velocity error. The error term, $\mathbf{s}$, is calculated using an exponential moving average calculated as

$$\mathbf{s}(t) = \begin{cases} 0, & t = 1 \\ \mathbf{e}'(1), & t = 2 \\ \alpha \mathbf{e}'(t-1) + (1-\alpha)\mathbf{s}(t-1), & t > 2 \end{cases} \tag{4.11}$$

where α is a positive, scalar smoothing factor and

$$\mathbf{e}'(t) = \mathbf{e}'_0 \sum_{i=2}^{N} a_{ij}(\mathbf{x}_1(t) - \mathbf{x}_i(t)),$$

where $\mathbf{e}'_0$ is a normalization factor vector. The inputs are normalized to be in the interval $[-1, 1]$. The exponential moving average is used to minimize sharp changes in the error which reduces jittering in the system.

The outputs of the fuzzy controller, K_{p_f}, K_{d_f} and κ_{1_f}, are scaling factors in the interval $[0, 1]$ and are used to calculate the tuning parameters by

$$\begin{aligned} K_p &= K_{p_f}(K_{p_{\max}} - K_{p_{\min}}) + K_{p_{\min}}, \\ K_d &= K_{d_f}(K_{d_{\max}} - K_{d_{\min}}) + K_{d_{\min}}, \\ \kappa_1 &= \kappa_{1_f}(\kappa_{1_{\max}} - \kappa_{1_{\min}}) + \kappa_{1_{\min}}, \end{aligned}$$

where $K_{p_{\max}}$, $K_{p_{\min}}$, $K_{d_{\max}}$,$K_{d_{\min}}$, $\kappa_{1_{\max}}$, and $\kappa_{1_{\min}}$ are the upper and lower limits of the respective controller parameters.

Remark 5. *In this chapter, it is assumed that each follower agent has access to the* *state information of the leader. Theoretically, the leader's signal would be transferred through intermediate follower agents. Practically, this is not realistic as the leader's signal delay would propagate as it passes through the follower agents. This issue is addressed later and the FLC methodology is improved in Chapter 5.*

Five membership functions are used for the input and output, resulting in a total of 75 rules. Fig. 4.1 shows the membership functions of the inputs and outputs for the fuzzy controller. Triangular membership functions are used to maintain the simplicity of the FLC. Other membership functions, such as Gaussian and trapezoidal types, were found to have a negligible impact on the overall performance. The rule base uses linguistic terms to simulate human judgment and is based on the user's control knowledge. The linguistic terms are defined in Table 4.1. Table 4.2 and 4.3 show the input-output rule bases for K_p and K_d and Table 4.4 shows the input-output rule base for κ_1. The most common method, center of area, is used for the defuzzification of the rule bases [33].

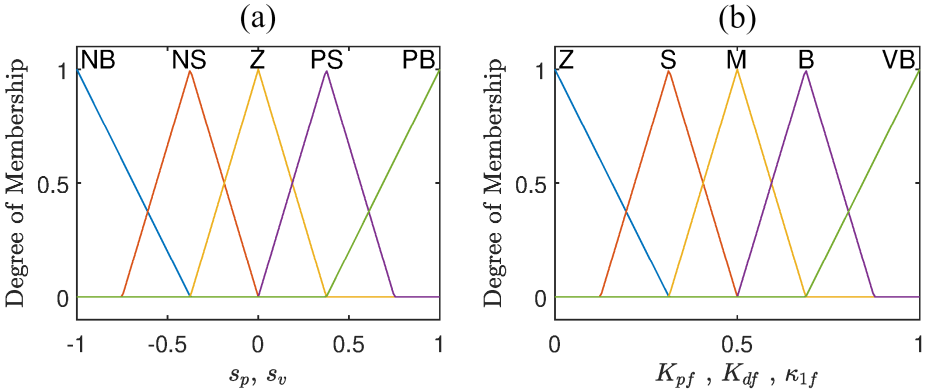

Figure 4.1: (a) Input membership functions and (b) output membership functions for PD+FLC

Table 4.1: Linguistic terms

N	Negative	S	Small
P	Positive	M	Medium
V	Very	B	Big
Z	Zero		

Table 4.2: Rule base for K_{p_f}

s_v/s_p	NB	NS	Z	PS	PB
NB	VB	VB	VB	VB	VB
NS	B	B	B	M	VB
Z	Z	Z	M	S	S
PS	B	B	B	M	VB
PB	VB	VB	VB	VB	VB

Table 4.3: Rule base for K_{d_f}

s_v/s_p	NB	NS	Z	PS	PB
NB	Z	S	M	M	VB
NS	S	B	M	VB	VB
Z	M	M	M	VB	VB
PS	B	VB	VB	VB	VB
PB	VB	VB	VB	VB	VB

Table 4.4: Rule base for κ_{1_f}

s_v/s_p	NB	NS	Z	PS	PB
NB	M	B	B	B	VB
NS	M	B	B	B	VB
Z	S	S	M	M	M
PS	B	B	B	M	VB
PB	VB	VB	VB	VB	VB

4.3 Simulations

This section describes the simulation parameters for synchronizing a networked group of DC motors, investigates the feasible ranges of gains for two network cases, and presents the simulation results of the performance improvement by the proposed controller.

4.3.1 Simulation Model

Groups of five and seven agents with the leader-follower communication topologies shown in Fig. 4.2 are simulated. The corresponding $\mathcal{A}^*$ and L_1^* matrices are constructed as

$$\mathcal{A}^a = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \end{bmatrix}, \; L_1^a = \begin{bmatrix} -3 & 1 & 0 & 1 \\ 1 & -3 & 1 & 0 \\ 0 & 1 & -2 & 1 \\ 1 & 0 & 1 & -2 \end{bmatrix},$$

$$\mathcal{A}^b = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 \end{bmatrix}, \; L_1^b = \begin{bmatrix} -4 & 1 & 1 & 1 & 0 & 0 \\ 1 & -4 & 0 & 0 & 1 & 1 \\ 1 & 0 & -2 & 1 & 0 & 0 \\ 1 & 0 & 1 & -2 & 0 & 0 \\ 0 & 1 & 0 & 0 & -2 & 1 \\ 0 & 1 & 0 & 0 & 1 & -2 \end{bmatrix},$$

where $*$ refers to each case in Fig. 4.2.

The system is modeled as in (4.1) and each agent has the dynamics of a second order linear DC motor represented as

$$\mathbf{x} = \begin{bmatrix} \theta \\ \dot{\theta} \end{bmatrix}, A = \begin{bmatrix} 0 & 1 \\ 0 & -\frac{b}{J} \end{bmatrix}, B = \begin{bmatrix} 0 \\ \frac{1}{J} \end{bmatrix}, C = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \tag{4.12}$$

where θ is the angular position, $\dot{\theta}$ is the angular velocity, $b = 0.05\ Nms$ is the damping coefficient, and $J = 0.007\ kgm^2$ is the moment of inertia. The leader is assigned the trajectory $\theta_1 = \sin(\frac{\pi t}{2})°$. The initial state of the leader is $\mathbf{x}_1 = \begin{bmatrix} 0, & 0 \end{bmatrix}^T$. The initial position of the followers is $\theta_{2i} = 28.7°, \theta_{3i} = 14.3°, \theta_{4i} = -14.3°, \theta_{5i} = -28.7°$, and the initial velocity of all followers is $\dot{\theta}_i = 0°/sec$. For the case of seven agents, $\theta_{6i} = 43.0°$ and $\theta^{\circ}_{7i} = -43.0°$. The local feedback gain is designed to have the poles $\mathbf{p} = \begin{bmatrix} 0, & -1 \end{bmatrix}$ which corresponds to a local gain matrix of $\mathbf{l} = \begin{bmatrix} 0, & 0.043 \end{bmatrix}$.

Each communication topology is simulated with a different measure of delays. Case (a) simulates the communication topology in Fig. 4.2 (a), subject to time-varying delays of $\bar{T}_{ij} = 0.02\ s$ and $\delta_{ij} = 0.003\ s$. Case (b) simulates the communication topology in Fig. 4.2 (b), subject to time-varying delays of $\bar{T}_{ij} = 0.03\ s$ and $\delta_{ij} = 0.003\ s$.

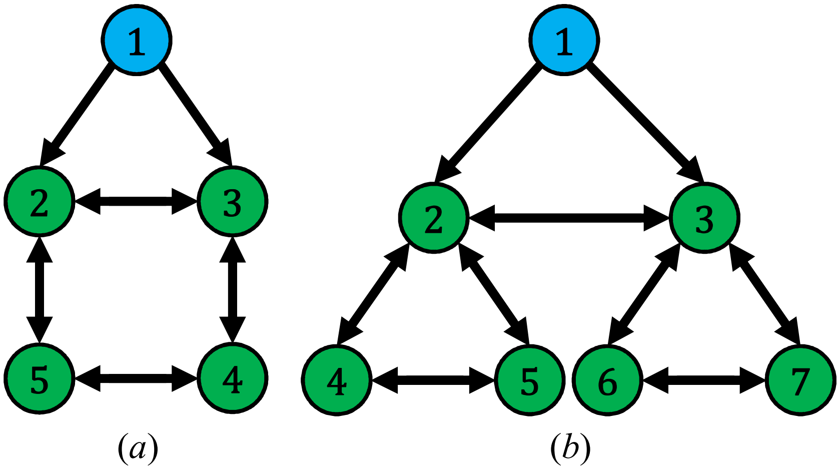

Figure 4.2: Leader-follower communication topology for linear simulations with (a) five agents (b) seven agents

4.3.2 Gain Feasibility Results

The range of feasible network gains are determined by Theorem 1 with $\gamma = 0.01$ and $S = 0.18\mathbf{I}_n$. The following figures show the feasibly ranges of K_p and K_d, where green represents the feasible range of gains and blue represents the infeasible range of gains.

From Figs. 4.3 - 4.6, it can be observed that as the maximum delay increases, the feasible range of gains is reduced. It can also be observed that having more complex communication topologies with more communication links can marginally increase the feasible range of gains for lower maximum delays.

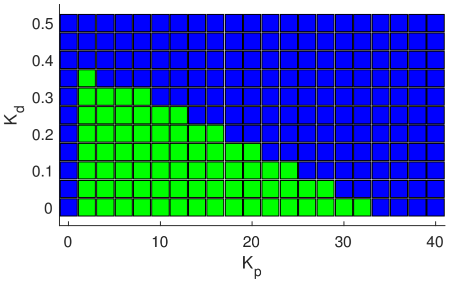

Figure 4.3: Feasibility range for topology (a) with $\bar{T}_{ij_{max}} = 0.023$ s

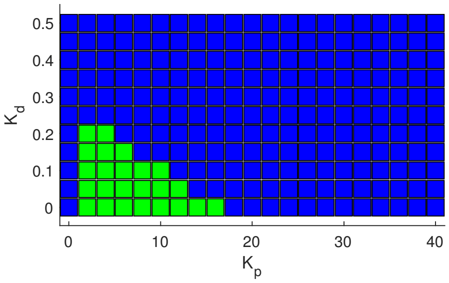

Figure 4.4: Feasibility range for topology (a) with $\bar{T}_{ij_{max}} = 0.033$ s

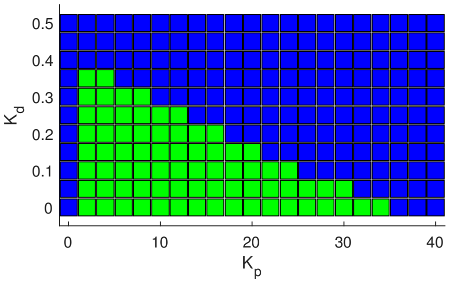

Figure 4.5: Feasibility range for topology (b) with $\bar{T}_{ij_{max}} = 0.023$ s

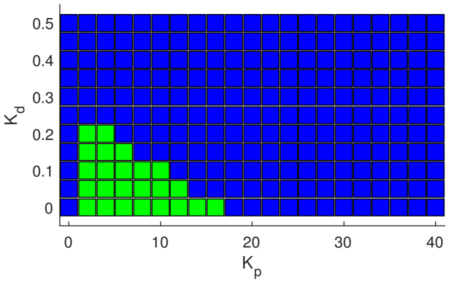

Figure 4.6: Feasibility range for topology (b) with $\bar{T}_{ij_{max}} = 0.033$ s

4.3.3 Simulation Results

First, the communication topology in Fig. 4.2 (a) is simulated with delays of $\bar{T}_{ij} = 0.02$ s and $\delta_{ij} = 0.003$ s. Fig. 4.7 shows the RMSE, as defined in Equation 3.1 in Chapter 3, for cases with time-varying gains, $\mathbf{K}$, and constant proportions of self-delay, κ_1. Figs. 4.8 - 4.10 show the position and position error of the simulations for cases of $\kappa_1 = 1.00$, $\kappa_1 = 0.75$, and $\kappa_1 = 0.50$. The RMSE values for each case are presented in Table 4.5.

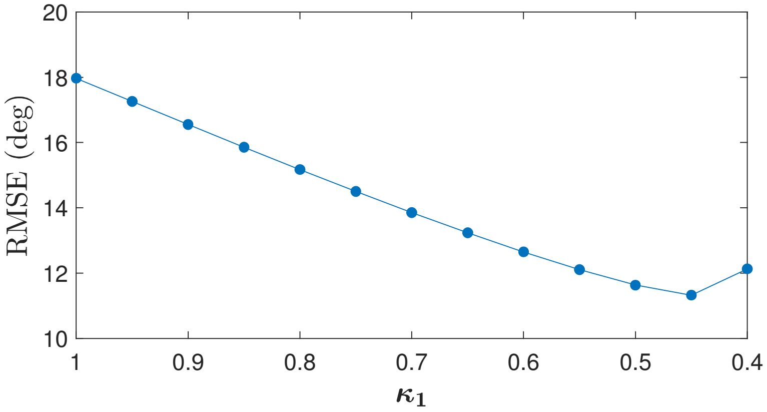

Figure 4.7: RMSE with various constant values of κ_1 for Case (a)

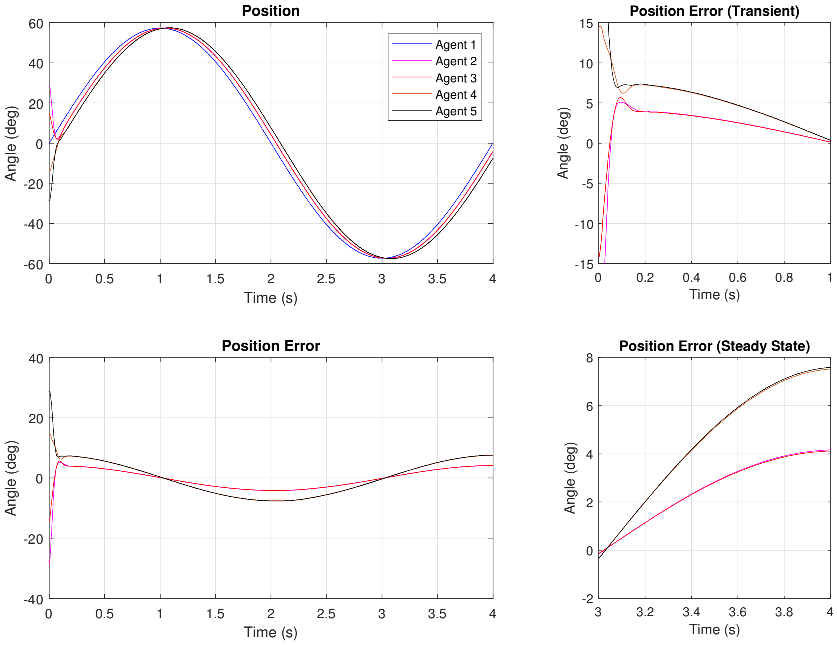

Figure 4.8: Agent position and position error for $\kappa_1 = 1.00$ for Case (a)

It can be seen from Figs. 4.7 - 4.10 and Table 4.5 that as κ_1 is reduced and the proportion of self-delay is increased, the RMSE is reduced. It is also observed in Fig. 4.7 that once κ_1 is lowered to a certain extent, $\kappa_1 = 0.45$, the RMSE begins to

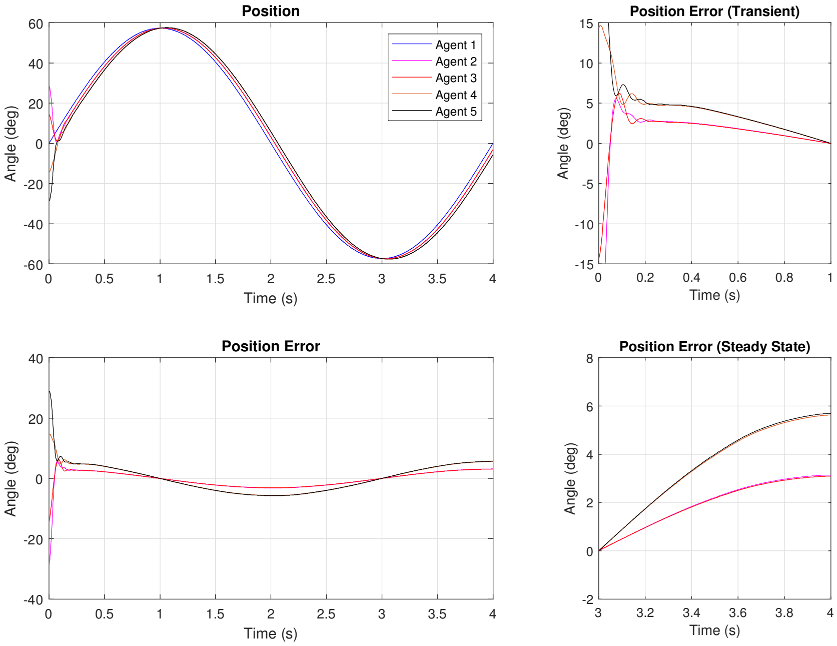

Figure 4.9: Agent position and position error for $\kappa_1 = 0.75$ for Case (a)

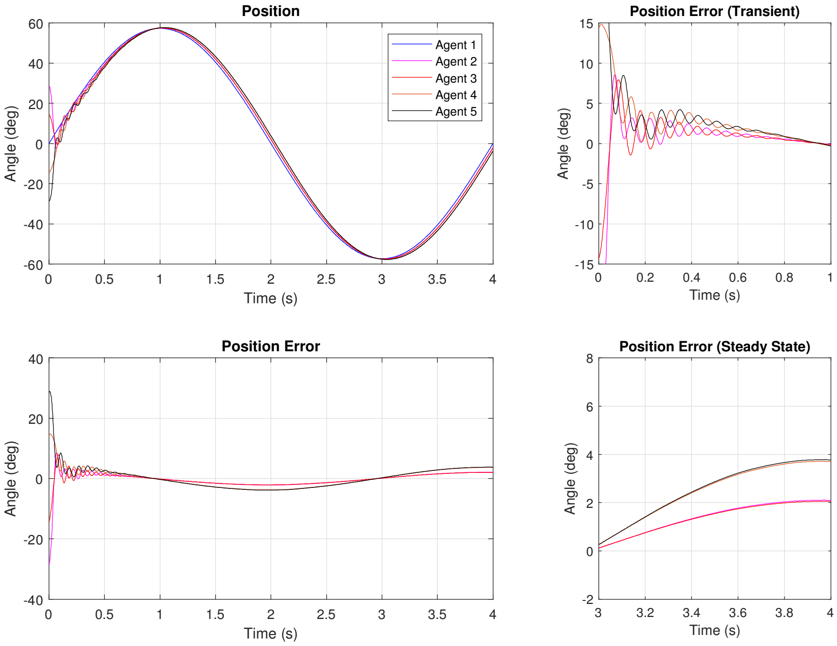

Figure 4.10: Agent position and position error for $\kappa_1 = 0.50$ for Case (a)

Table 4.5: Simulation performance metrics for Cases (a) and (b)

	RMSE	
	Case (a)	Case (b)
$\kappa_1 = 1.00$	18.0°	55.8°
$\kappa_1 = 0.75$	14.5°	45.2°
$\kappa_1 = 0.50$	11.6°	35.2°
FLC κ_1	11.2°	34.1°

Table 4.6: FLC parameters for Cases (a) and (b)

	Case (a)	Case (b)
$\mathbf{e}'_0$	$\begin{bmatrix}3.33 & 2.00\end{bmatrix}$	$\begin{bmatrix}0.83 & 0.5\end{bmatrix}$
α	0.98	0.98
$\kappa_{1_{\max}}$	0.7	0.7
$\kappa_{1_{\min}}$	0	0
$K_{p_{\max}}$	10	4
$K_{p_{\min}}$	5	2
$K_{d_{\max}}$	0.3	0.2
$K_{d_{\min}}$	0	0

increase. The RMSE begins to increase exponentially for values of κ_1 below 0.4 and are excluded from Fig. 4.7. In Fig. 4.10, for $\kappa_1 = 0.5$, oscillations appear during the first 0.5 s in the trajectories of the follower agents.

The case where κ_1 is tuned with FLC is shown in Fig. 4.11 and the scaling factors, K_{p_f}, K_{d_f} and κ_{1_f} are shown in Fig. 4.12. The controller parameters are presented in Table 4.6. The FLC mixed feedback controller reduced the RMSE below that of the case where $\kappa_1 = 0.5$, while minimizing the undesirable oscillations. For Case (a), the proposed controller has a 37.6% improved error performance from the case where $\kappa_1 = 1.00$ (the feedback contains no self-delay) and a 22.6% improved error performance from the case where $\kappa_1 = 0.75$, which has the equivalent performance with regard to the oscillations.

Next, the communication topology in Fig. 4.2 (b) is simulated with higher delays of $\bar{T}_{ij} = 0.03$ and $\delta_{ij} = 0.003$. Fig. 4.13 shows the RMSE for cases with time-varying gains, $\mathbf{K}$, and constant proportions of self-delay, κ_1. Fig. 4.14 - 4.16 show the position

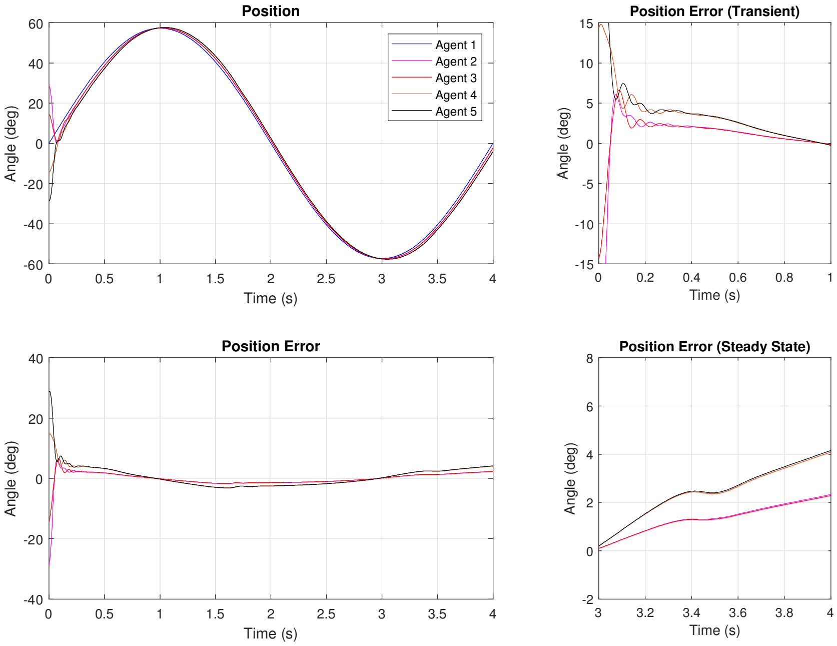

Figure 4.11: Agent position and position error for FLC tuned κ_1 for Case (a)

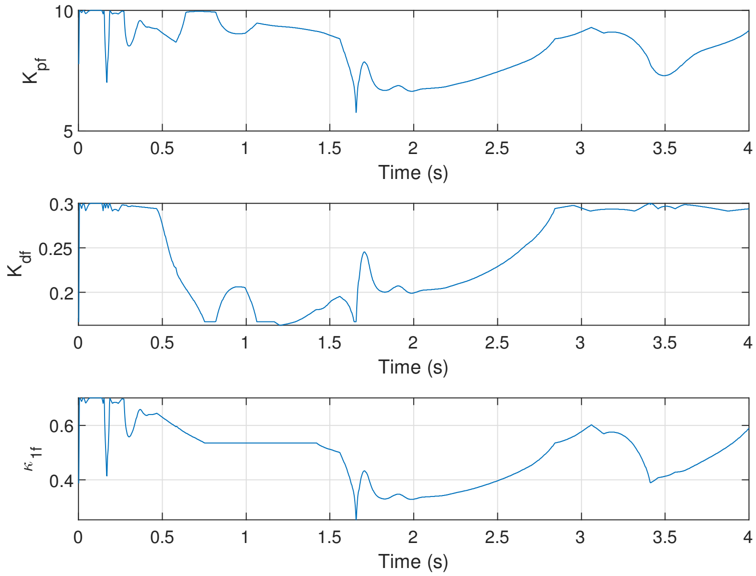

Figure 4.12: Time-varying K_p, K_d and κ_1 tuned with FLC for Case (a)

and position error of the simulations for cases of $\kappa_1 = 1.00$, $\kappa_1 = 0.75$, and $\kappa_1 = 0.50$. The RMSE values for each case are presented in Table 4.5. The case where κ_1 is tuned with FLC is shown in Fig. 4.17 and the scaling factors, K_{p_f}, K_{d_f} and κ_{1_f} are shown in Fig. 4.18. The controller parameters are presented in Table 4.6.

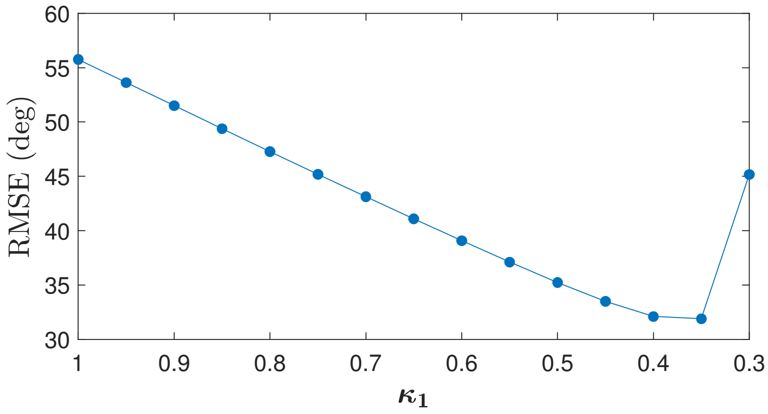

Figure 4.13: RMSE with various constant values of κ_1 for Case (b)

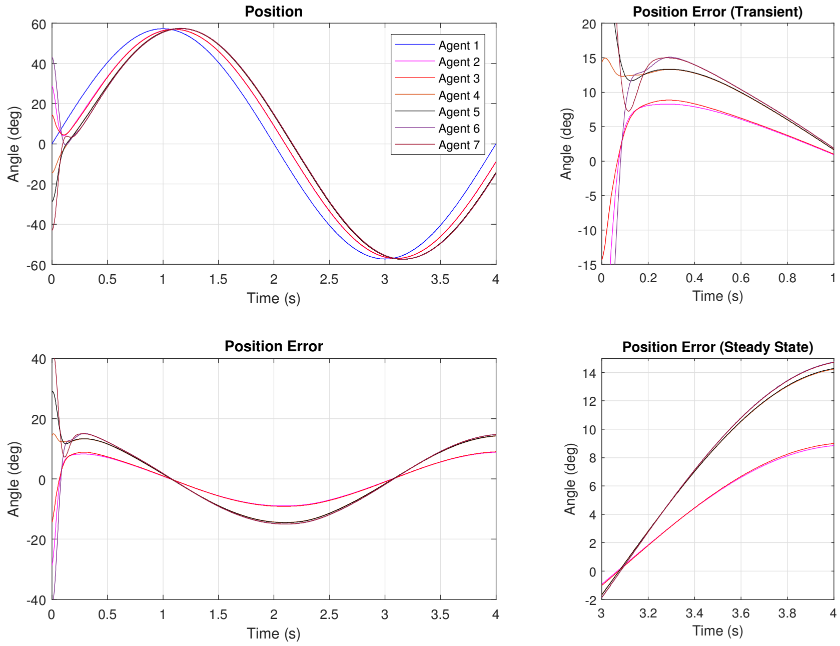

Figure 4.14: Agent position and position error for $\kappa_1 = 1.00$ for Case (b)

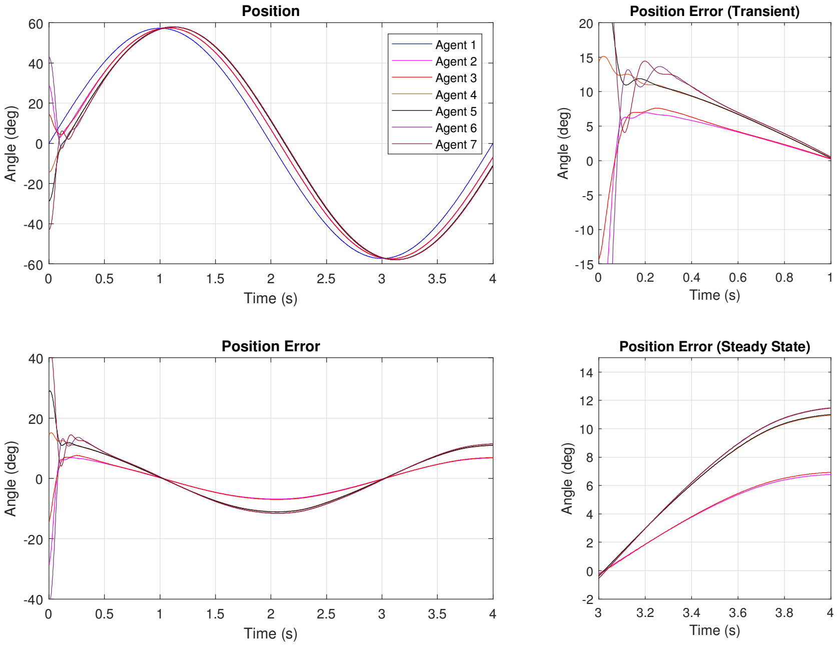

Figure 4.15: Agent position and position error for $\kappa_1 = 0.75$ for Case (b)

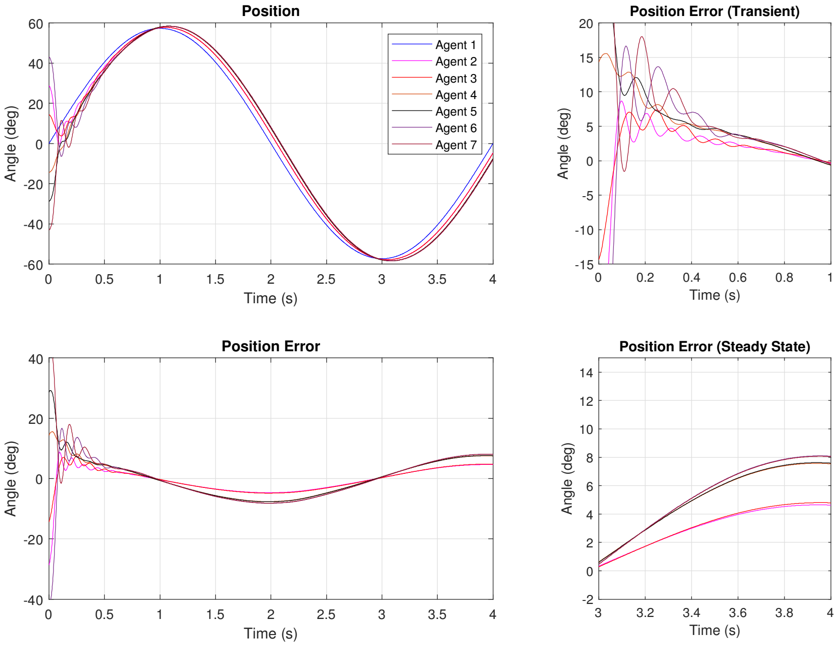

Figure 4.16: Agent position and position error for $\kappa_1 = 0.50$ for Case (b)

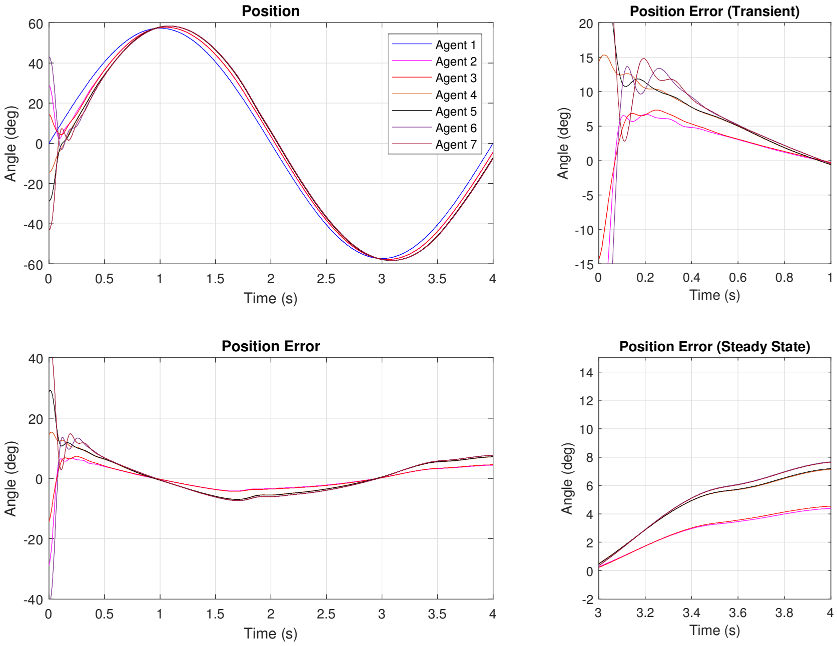

Figure 4.17: Agent position and position error for FLC tuned κ_1 for Case (b)

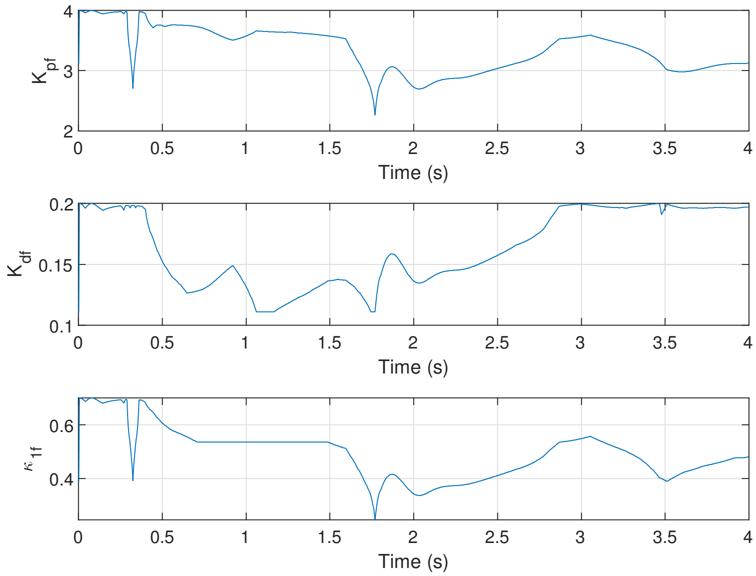

Figure 4.18: Time-varying K_p, K_d and κ_1 tuned with FLC for Case (b)

In comparison to Case (a), the ranges of feasible gains, K_p and K_d, are lower, which is attributed to the increased delays. The increased delays, higher number of agents, and lower feasible gains results in higher RMSE values for Case (b), compared to Case (a). For Case (b), the proposed controller has a 38.8% improved error performance from the case where $\kappa_1 = 1.00$ (the feedback contains no self-delay) and a 24.4% improved error performance from the case where $\kappa_1 = 0.75$, which has the equivalent performance with regard to the oscillations. The percentage of performance improvement is similar for both of the simulated Cases, (a) and (b). The simulations show that the proposed method of improving the consensus performance is significant and offers room for improvement based on the concept of using online tuned time-varying gains in the control policy.

4.4 Summary

An FLC method for tuning the gains and proportions of self-delay in the controller feedback has been developed for linear systems. Simulations of a group of DC motors show that for networks with higher delays, the feasible range of gains that are derived from the proposed LMI, is reduced. Increasing the proportion of self-delay in the controller feedback improves the overall synchronization tracking performance up to an extent before the performance begins to decrease. Simulations show that the proposed FLC policy improves the tracking performance while minimizing undesirable oscillations for five and seven agent linear MASs.

Chapter 5

Improving Performance for Euler-Lagrange MASs

This chapter introduces the problem framework of synchronizing a networked group of EL agents in the presence of asynchronous time-varying delays. The error dynamics for nonlinear mixed-type feedback with time-varying parameters are derived. The proposed NTSMC policy is presented and the FLC method from the preceding chapter is improved upon. The stability of the system is proven using Lyapunov criteria. Simulations and experiments of a group of Phantom Omni manipulators are conducted to demonstrate an improvement in synchronization tracking performance and to validate the proposed controller.

5.1 Problem Formulation

Consider a MAS composed of $\mathbb{N}$ homogeneous EL agents, represented by the following dynamics

$$M_i(\boldsymbol{q}_i)\ddot{\boldsymbol{x}}_i + C_i(\boldsymbol{q}_i, \dot{\boldsymbol{q}}_i)\dot{\boldsymbol{x}}_i + \boldsymbol{g}_i(\boldsymbol{q}_i) = \boldsymbol{u}_i + \boldsymbol{f}_i, \quad i \in \mathbb{N} \tag{5.1}$$

where the parameters are defined as in (3.14).

Remark 6. *In this chapter the followers are represented by $i = \{1, \ldots, \mathbb{N}\}$ and a single virtual leader is represented by $i = 0$.*

Property 1. *For all EL-type agents, M_i is symmetric positive definite and $\dot{M}_i - 2C_i$ is skew-symmetric.*

Assumption 3. *The communication networks considered in this chapter are directed trees that have the leader agent as the root node. That is, the leader node has a series of connected edges to every other node.*

Assumption 4. *The upper bound of the forces due to disturbances and frictional forces is known and defined as B_f, where $B_f \geq \|\boldsymbol{f}_i\|$.*

5.2 Control Development

In this section, the mixed-type error dynamics are derived, an NTSMC policy is proposed, the stability and error bounds are analyzed using Lyapunov conditions and Schwarz inequalities, and an improved FLC system is developed to tune the controller parameters as a function of the MAS error.

5.2.1 Error Dynamics

The mixed-type error dynamics of the system can be defined as

$$\boldsymbol{e}_i = \boldsymbol{\varpi}_{1_i}(t)\boldsymbol{E}_{1_i}(t) + \boldsymbol{\varpi}_{2_i}(t)\boldsymbol{E}_{2_i}(t), \tag{5.2}$$

where $\boldsymbol{\varpi}_{1_i} \in \mathbb{R}^{3\times3}$ and $\boldsymbol{\varpi}_{2_i} \in \mathbb{R}^{3\times3}$ are positive diagonal matrices of the respective weights for the proportions of feedback without and with self-delay. The parameters, $\boldsymbol{\varpi}_{1_i}$ and $\boldsymbol{\varpi}_{2_i}$, satisfy $\boldsymbol{\varpi}_{1_i} + \boldsymbol{\varpi}_{2_i} = I_3$, where I_3 represents the 3 by 3 identity matrix. The delayed error components in (5.2), $\boldsymbol{E}_{1_i}(t) \in \mathbb{R}^3$ and $\boldsymbol{E}_{2_i}(t) \in \mathbb{R}^3$, are defined as

$$\boldsymbol{E}_{1_i}(t) = \sum_{j=1}^{\mathbb{N}} a_{ij}[\boldsymbol{x}_i(t) - \boldsymbol{x}_j(t - T_{ij}(t))] + b_i[\boldsymbol{x}_i(t) - \boldsymbol{x}_0(t - T_{i0}(t))], \tag{5.3}$$

$$\boldsymbol{E}_{2_i}(t) = \sum_{j=1}^{\mathbb{N}} a_{ij}[\boldsymbol{x}_i(t - \bar{T}) - \boldsymbol{x}_j(t - T_{ij}(t))] + b_i[\boldsymbol{x}_i(t - \bar{T}) - \boldsymbol{x}_0(t - T_{i0}(t))]. \tag{5.4}$$

The derivative of the error dynamics is given as

$$\dot{\boldsymbol{e}}_i = \boldsymbol{E}_{pv_i}(t) + \boldsymbol{\varpi}_{1_i}(t)\dot{\boldsymbol{E}}_{1_i}(t) + \boldsymbol{\varpi}_{2_i}(t)\dot{\boldsymbol{E}}_{2_i}(t), \tag{5.5}$$

where

$$\boldsymbol{E}_{pv_i}(t) = \dot{\boldsymbol{\varpi}}_{1_i}(t)\boldsymbol{E}_{1_i}(t) + \dot{\boldsymbol{\varpi}}_{2_i}(t)\boldsymbol{E}_{2_i}(t), \tag{5.6}$$

$$\dot{\boldsymbol{E}}_{1_i}(t) = \sum_{j=1}^{\mathbb{N}} a_{ij}[\dot{\boldsymbol{x}}_i(t) - \dot{\boldsymbol{x}}_j(t - T_{ij}(t))\dot{T}_{ij}(t)] + b_i[\dot{\boldsymbol{x}}_i(t) - \dot{\boldsymbol{x}}_0(t - T_{i0}(t))\dot{T}_{i0}(t)], \tag{5.7}$$

$$\dot{\boldsymbol{E}}_{2_i}(t) = \sum_{j=1}^{\mathbb{N}} a_{ij}[\dot{\boldsymbol{x}}_i(t - \bar{T}) - \dot{\boldsymbol{x}}_j(t - T_{ij}(t))\dot{T}_{ij}(t)]$$

$$+ b_i[\dot{\boldsymbol{x}}_i(t-\bar{T}) - \dot{\boldsymbol{x}}_0(t-T_{i0}(t))\dot{T}_{i0}(t)]. \tag{5.8}$$

Remark 7. *In many applications, the velocity and acceleration of the system are not available. The velocity and acceleration can be estimated as the first and second time derivatives of the position data. Here, the neighbours' velocities are considered as the time derivative of the transmitted position signal, $\boldsymbol{x}_j(t-T_{ij}(t))$, therefore*

$$\dot{\boldsymbol{x}}_j(t-T_{ij}(t))\dot{T}_{ij}(t) = \frac{\partial(\boldsymbol{x}_j(t-T_{ij}(t)))}{\partial t}, \tag{5.9}$$

and $\dot{T}_{ij}(t)$ is not required to be known.

Following Remark 7, (5.7) and (5.8) may be rewritten as

$$\dot{\boldsymbol{E}}_{1_i}(t) = \sum_{j=1}^{\mathbb{N}} a_{ij}\Big[\dot{\boldsymbol{x}}_i(t) - \frac{\partial(\boldsymbol{x}_j(t-T_{ij}(t)))}{\partial t}\Big] + b_i\Big[\dot{\boldsymbol{x}}_i(t) - \frac{\partial(\boldsymbol{x}_0(t-T_{i0}(t)))}{\partial t}\Big], \tag{5.10}$$

$$\dot{\boldsymbol{E}}_{2_i}(t) = \sum_{j=1}^{\mathbb{N}} a_{ij}\Big[\dot{\boldsymbol{x}}_i(t-\bar{T}) - \frac{\partial(\boldsymbol{x}_j(t-T_{ij}(t)))}{\partial t}\Big] + b_i\Big[\dot{\boldsymbol{x}}_i(t-\bar{T}) - \frac{\partial(\boldsymbol{x}_0(t-T_{i0}(t)))}{\partial t}\Big]. \tag{5.11}$$

The second-order derivative of the error dynamics is given as

$$\ddot{\boldsymbol{e}}_i = \boldsymbol{E}_{va_i}(t) + \boldsymbol{\varpi}_{1_i}(t)\ddot{\boldsymbol{E}}_{1_i}(t) + \boldsymbol{\varpi}_{2_i}(t)\ddot{\boldsymbol{E}}_{2_i}(t), \tag{5.12}$$

where

$$\boldsymbol{E}_{va_i}(t) = \ddot{\boldsymbol{\varpi}}_{1_i}(t)\boldsymbol{E}_{1_i}(t) + \ddot{\boldsymbol{\varpi}}_{2_i}(t)\boldsymbol{E}_{2_i}(t) + 2\dot{\boldsymbol{\varpi}}_{1_i}(t)\dot{\boldsymbol{E}}_{1_i}(t) + 2\dot{\boldsymbol{\varpi}}_{2_i}(t)\dot{\boldsymbol{E}}_{2_i}(t), \tag{5.13}$$

$$\ddot{\boldsymbol{E}}_{1_i}(t) = \sum_{j=1}^{\mathbb{N}} a_{ij}\Big[\ddot{\boldsymbol{x}}_i(t) - \frac{\partial^2(\boldsymbol{x}_j(t-T_{ij}(t)))}{\partial t^2}\Big] + b_i\Big[\ddot{\boldsymbol{x}}_i(t) - \frac{\partial^2(\boldsymbol{x}_0(t-T_{i0}(t)))}{\partial t^2}\Big], \tag{5.14}$$

$$\ddot{\boldsymbol{E}}_{2_i}(t) = \sum_{j=1}^{\mathbb{N}} a_{ij}\Big[\ddot{\boldsymbol{x}}_i(t-\bar{T}) - \frac{\partial^2(\boldsymbol{x}_j(t-T_{ij}(t)))}{\partial t^2}\Big] + b_i\Big[\ddot{\boldsymbol{x}}_i(t-\bar{T}) - \frac{\partial^2(\boldsymbol{x}_0(t-T_{i0}(t)))}{\partial t^2}\Big]. \tag{5.15}$$

Remark 8. *In practice, the terms $\dot{\boldsymbol{x}}$, $\ddot{\boldsymbol{x}}$, $\dot{\boldsymbol{\varpi}}_1$, and $\ddot{\boldsymbol{\varpi}}_1$ are obtained by numerically approximating the time-derivatives with finite differences, which can result in rapid changes in values. First order low-pass filters are applied to the terms to minimize issues associated with measurement noises and chatterings.*

5.2.2 Controller Design

As in [52], the sliding surface is constructed from the position error and the velocity error as follows

$$\boldsymbol{s}_i = \boldsymbol{e}_i + \beta(\dot{\boldsymbol{e}}_i)^{\alpha}, \tag{5.16}$$

$$\dot{\boldsymbol{s}}_i = \dot{\boldsymbol{e}}_i + \alpha\beta(\dot{\boldsymbol{e}}_i)^{\alpha-1}\ddot{\boldsymbol{e}}_i, \tag{5.17}$$

where $\beta > 0$. $\alpha = p/q$ with $p > 0$, $q > 0$ and p, q are adjacent odd numbers such that $1 < \alpha < 2$. The controller for the followers is designed as

$$\begin{aligned} \boldsymbol{u}_i =& C_i\dot{\boldsymbol{x}}_i + \boldsymbol{g}_i + \boldsymbol{\varpi}_{1_i}^{-1}(t)\Big(\sum_{j=1}^{\mathbb{N}} a_{ij} + b_i\Big)^{-1} M_i\Bigg\{ - \frac{(\dot{\boldsymbol{e}}_i)^{(2-\alpha)}}{\alpha\beta} \\ & + \boldsymbol{\varpi}_{1_i}(t)\sum_{j=1}^{\mathbb{N}} a_{ij}\frac{\partial^2(\boldsymbol{x}_j(t - T_{ij}(t)))}{\partial t^2} + \boldsymbol{\varpi}_{1_i}(t)b_i\frac{\partial^2(\boldsymbol{x}_0(t - T_{i0}(t)))}{\partial t^2} \\ & - \boldsymbol{\varpi}_{2_i}(t)\ddot{\boldsymbol{E}}_{2_i}(t) - \boldsymbol{E}_{va_i}(t) - \boldsymbol{\kappa}_i(t)\text{sgn}(\boldsymbol{s}_i)\Bigg\} - B_f\text{sgn}(\boldsymbol{s}_i), \end{aligned} \tag{5.18}$$

where the control gain, $\boldsymbol{\kappa}_i \in \mathbb{R}^{3\times 3}$, is a positive diagonal matrix for the ith agent. Lemma 1 [49] is introduced for the subsequent finite-time stability proof.

Lemma 1. *For a non-Lipschitz continuous nonlinear system $\dot{x} = f(x)$, suppose there exists a continuous function $V(x)$ defined on a neighbourhood of the origin. If $V(x)$ is positive definite and there exist real numbers $c > 0$ and $0 < \gamma < 1$ such that $\dot{V}(x) + cV^{\gamma} \leq 0$, then the origin is locally finite-time stable, and the settling time, contingent on the initial state $x(0) = x_0$, will satisfy:*

$$T(x_0) \leq \frac{V(x_0)^{1-\gamma}}{c(1-\gamma)}.$$

5.2.3 Stability Analysis

In the following section, the stability of a single agent is proven. Considering that the MAS is fully distributed, the stability of a single agent, including transmitted time delayed signals from neighbours, verifies the stability of the entire MAS [56]. By combining (5.1) and (5.18), the closed-loop dynamics of the ith agent can be represented as

$$\begin{aligned} M_i\ddot{\boldsymbol{x}}_i + &C_i\dot{\boldsymbol{x}}_i + \boldsymbol{g}_i = \boldsymbol{f}_i - B_f\mathrm{sgn}(\boldsymbol{s}_i) + C_i\dot{\boldsymbol{x}}_i + \boldsymbol{g}_i \\ &+ \boldsymbol{\varpi}_{1_i}^{-1}(t)\Big(\sum_{j=1}^{\mathbb{N}} a_{ij} + b_i\Big)^{-1} M_i\Bigg\{ -\frac{(\dot{\boldsymbol{e}}_i)^{(2-\alpha)}}{\alpha\beta} + \boldsymbol{\varpi}_{1_i}(t)\sum_{j=1}^{\mathbb{N}} a_{ij}\frac{\partial^2(\boldsymbol{x}_j(t-T_{ij}(t)))}{\partial t^2} \\ &+ \boldsymbol{\varpi}_{1_i}(t)b_i\frac{\partial^2(\boldsymbol{x}_0(t-T_{i0}(t)))}{\partial t^2} - \boldsymbol{\varpi}_{2_i}(t)\ddot{\boldsymbol{E}}_{2_i}(t) - \boldsymbol{E}_{va_i}(t) - \boldsymbol{\kappa}_i(t)\mathrm{sgn}(\boldsymbol{s}_i)\Bigg\}, \end{aligned} \tag{5.19}$$

which can be simplified and rearranged into

$$\begin{aligned} \boldsymbol{\varpi}_{1_i}(t)\Big(\sum_{j=1}^{\mathbb{N}} a_{ij} + b_i\Big)\ddot{\boldsymbol{x}}_i - \boldsymbol{\varpi}_{1_i}(t)\sum_{j=1}^{\mathbb{N}} a_{ij}\frac{\partial^2(\boldsymbol{x}_j(t-T_{ij}(t)))}{\partial t^2} \\ - \boldsymbol{\varpi}_{1_i}(t)b_i\frac{\partial^2(\boldsymbol{x}_0(t-T_{i0}(t)))}{\partial t^2} + \boldsymbol{\varpi}_{2_i}(t)\ddot{\boldsymbol{E}}_{2_i}(t) + \boldsymbol{E}_{va_i}(t) \\ = \boldsymbol{\varpi}_{1_i}(t)\Big(\sum_{j=1}^{\mathbb{N}} a_{ij} + b_i\Big)(\boldsymbol{f}_i - B_f\mathrm{sgn}(\boldsymbol{s}_i)) - \frac{(\dot{\boldsymbol{e}}_i)^{(2-\alpha)}}{\alpha\beta} - \boldsymbol{\kappa}_i(t)\mathrm{sgn}(\boldsymbol{s}_i). \end{aligned} \tag{5.20}$$

Substituting (5.12) and (5.14) into (5.20) gives

$$\begin{aligned} \boldsymbol{\varpi}_{1_i}(t)\ddot{\boldsymbol{E}}_{1_i}(t) + \boldsymbol{\varpi}_{2_i}(t)\ddot{\boldsymbol{E}}_{2_i}(t) + \boldsymbol{E}_{va_i}(t) = \ddot{\boldsymbol{e}}_i = &\boldsymbol{\varpi}_{1_i}(t)\Big(\sum_{j=1}^{\mathbb{N}} a_{ij} + b_i\Big)(\boldsymbol{f}_i - B_f\mathrm{sgn}(\boldsymbol{s}_i)) \\ &- \frac{(\dot{\boldsymbol{e}}_i)^{(2-\alpha)}}{\alpha\beta} - \boldsymbol{\kappa}_i(t)\mathrm{sgn}(\boldsymbol{s}_i). \end{aligned} \tag{5.21}$$

Multiplying both sides by $\alpha\beta\mathrm{diag}((\dot{\boldsymbol{e}}_i)^{\alpha-1})$ gives

$$\dot{\boldsymbol{e}}_i + \alpha\beta\mathrm{diag}((\dot{\boldsymbol{e}}_i)^{\alpha-1})\ddot{\boldsymbol{e}}_i = \alpha\beta\boldsymbol{\varpi}_{1_i}(t)\Big(\sum_{j=1}^{\mathbb{N}} a_{ij} + b_i\Big)\mathrm{diag}((\dot{\boldsymbol{e}}_i)^{\alpha-1})(\boldsymbol{f}_i - B_f\mathrm{sgn}(\boldsymbol{s}_i))$$

$$-\boldsymbol{\kappa}_i(t)\alpha\beta\mathrm{diag}((\dot{\boldsymbol{e}}_i)^{\alpha-1})\mathrm{sgn}(\boldsymbol{s}_i). \tag{5.22}$$

Substituting (5.17) into (5.22) gives

$$\begin{aligned}\dot{\boldsymbol{s}}_i =&\alpha\beta\boldsymbol{\varpi}_{1_i}(t)\Big(\sum_{j=1}^{\mathbb{N}} a_{ij}+b_i\Big)\mathrm{diag}((\dot{\boldsymbol{e}}_i)^{\alpha-1})(\boldsymbol{f}_i-B_f\mathrm{sgn}(\boldsymbol{s}_i))\\ &-\boldsymbol{\kappa}_i(t)\alpha\beta\mathrm{diag}((\dot{\boldsymbol{e}}_i)^{\alpha-1})\mathrm{sgn}(\boldsymbol{s}_i).\end{aligned} \tag{5.23}$$

Consider the following Lyapunov function

$$V_s=\frac{1}{2}\boldsymbol{s}_i^T\boldsymbol{s}_i.$$

The derivative of the Lyapunov function is

$$\begin{aligned}\dot{V}_s=\boldsymbol{s}_i^T\dot{\boldsymbol{s}}_i =&\alpha\beta\boldsymbol{\varpi}_{1_i}(t)\Big(\sum_{j=1}^{\mathbb{N}} a_{ij}+b_i\Big)\boldsymbol{s}_i^T\mathrm{diag}((\dot{\boldsymbol{e}}_i)^{\alpha-1})(\boldsymbol{f}_i-B_f\mathrm{sgn}(\boldsymbol{s}_i))\\ &-\boldsymbol{\kappa}_i(t)\alpha\beta\mathrm{diag}((\dot{\boldsymbol{e}}_i)^{\alpha-1})\boldsymbol{s}_i^T\mathrm{sgn}(\boldsymbol{s}_i).\end{aligned} \tag{5.24}$$

Since $\boldsymbol{\kappa}_i>0$, $\mathrm{diag}((\dot{\boldsymbol{e}}_i)^{\alpha-1})\geq 0$, $\boldsymbol{s}_i^T\mathrm{sgn}(\boldsymbol{s}_i)\geq 0$, and $B_f\geq\|\boldsymbol{f}_i\|$, we have

$$\boldsymbol{s}_i^T\mathrm{diag}((\dot{\boldsymbol{e}}_i)^{\alpha-1})(\boldsymbol{f}_i-B_f\mathrm{sgn}(\boldsymbol{s}_i))\leq 0. \tag{5.25}$$

Therefore,

$$\begin{aligned}\dot{V}_s&\leq-\boldsymbol{\kappa}_i(t)\alpha\beta\mathrm{diag}((\dot{\boldsymbol{e}}_i)^{\alpha-1})\boldsymbol{s}_i^T\mathrm{sgn}(\boldsymbol{s}_i)\\ &\leq-\sqrt{2}\alpha_i\beta_i\boldsymbol{\kappa}_i(t)(\mathbf{e}_i^v)^{(\alpha_i-1)}(\frac{1}{2}\mathbf{s}_i^T\mathbf{s}_i)^{\frac{1}{2}}.\end{aligned} \tag{5.26}$$

Considering Lemma 1 and (5.26), we have $c=\sqrt{2}\alpha_i\beta_i\boldsymbol{\kappa}_i(t)(\|\mathbf{e}_i^v\|)^{(\alpha_i-1)}$ and $\gamma=\frac{1}{2}$. Therefore, we can conclude that it takes a finite time to reach the sliding surface.

Once the sliding surface has converged to 0, $\boldsymbol{s}_i=0$ and (5.16) gives

$$\dot{\boldsymbol{e}}_i=-\beta^{-\frac{1}{\alpha}}(\boldsymbol{e}_i)^{\frac{1}{\alpha}}. \tag{5.27}$$

To prove that the tracking error, $\boldsymbol{e}$, converges to 0, consider the Lyapunov function as

$$V_i^e = \frac{1}{2}(\boldsymbol{e}_i)^T \boldsymbol{e}_i. \tag{5.28}$$

Then

$$\dot{V}_i^e = (\boldsymbol{e}_i)^T \dot{\boldsymbol{e}}_i = (\boldsymbol{e}_i)^T[-\beta^{-\frac{1}{\alpha}}(\boldsymbol{e}_i)^{\frac{1}{\alpha}}] = -\beta^{-\frac{1}{\alpha}}(\|\boldsymbol{e}_i\|_1)^{1+\frac{1}{\alpha}} \leq 0. \tag{5.29}$$

Again, using Lemma 1, it can be concluded that the tracking error can converge to zero in a finite time.

Remark 9. *The tracking error, $\mathbf{e}_i$, refers to (5.2), which is the mixed-type delayed tracking error and contains the delayed signals of the agents.*

5.2.4 Error Boundedness

The convergence of the total mixed type error, $\boldsymbol{e}_i = \mathbf{0}$ has been proven to be achieved in a finite time, but we cannot conclude that $\boldsymbol{E}_{1_i}(t)$ and $\boldsymbol{E}_{2_i}(t)$ can converge to zero. From (5.2), we have

$$\boldsymbol{E}_{2_i}(t) = -\frac{\boldsymbol{\varpi}_{1_i}(t)}{\boldsymbol{\varpi}_{2_i}(t)}\boldsymbol{E}_{1_i}(t), \tag{5.30}$$

From (5.3) and (5.4), we have

$$\boldsymbol{E}_{1_i}(t) - \boldsymbol{E}_{2_i}(t) = (\sum_{j=1}^{\mathbb{N}} a_{ij} + b_i)[\boldsymbol{x}_i(t) - \boldsymbol{x}_i(t - \bar{T})]. \tag{5.31}$$

Therefore,

$$\boldsymbol{E}_{1_i}(t) = \boldsymbol{\varpi}_{2_i}(t)(\sum_{j=1}^{\mathbb{N}} a_{ij} + b_i)[\boldsymbol{x}_i(t) - \boldsymbol{x}_i(t - \bar{T})]. \tag{5.32}$$

Using the Schwarz inequality results in

$$\begin{aligned}
\|\boldsymbol{E}_{1_i}(t)\| &= \boldsymbol{\varpi}_{2_i}(t)(\sum_{j=1}^{\mathbb{N}} a_{ij} + b_i)\|\boldsymbol{x}_i(t) - \boldsymbol{x}_i(t - \bar{T})\| \\
&= \boldsymbol{\varpi}_{2_i}(t)(\sum_{j=1}^{\mathbb{N}} a_{ij} + b_i)\Big\| \int_0^{\bar{T}} \dot{\boldsymbol{x}}_i(\theta) d\theta \Big\| \\
&\leq \boldsymbol{\varpi}_{2_i}(t)(\sum_{j=1}^{\mathbb{N}} a_{ij} + b_i)\Big\| \bar{T} \int_0^{\bar{T}} |\dot{\boldsymbol{x}}_i(t-\theta)|^2 d\theta \Big\|^{\frac{1}{2}}
\end{aligned}$$

$$\le \boldsymbol{\varpi}_{2_i}(t)(\sum_{j=1}^{\mathbb{N}} a_{ij} + b_i)\bar{T}^{\frac{1}{2}}\|\dot{\boldsymbol{x}}_i\|. \tag{5.33}$$

Similarly,

$$\|\boldsymbol{E}_{2_i}(t)\| \le \boldsymbol{\varpi}_{1_i}(t)(\sum_{j=1}^{\mathbb{N}} a_{ij} + b_i)\bar{T}^{\frac{1}{2}}\|\dot{\boldsymbol{x}}_i\|. \tag{5.34}$$

Therefore, in the case where the total mixed type error, $\boldsymbol{e}_i$, converges to 0 and $\boldsymbol{E}_{1_i} \neq \boldsymbol{E}_{2_i} \neq 0$, the errors, $\boldsymbol{E}_{1_i}$ and $\boldsymbol{E}_{2_i}$, are bounded by (5.33) and (5.34). The error bounds are dependent on the network delays, proportion of self-delayed feedback, and the agent's state velocity.

5.2.5 Fuzzy Logic Control

Fuzzy logic tuning is applied to the gain parameter, $\boldsymbol{\kappa}_i$, and the weight of self-delay, $\boldsymbol{\varpi}_{1_i}$, in the NTSMC policy. FLC is applied here to adapt the gain to compensate for disturbances, encourage the agents' states to reach the sliding surface in the presence of time-vary delays, and avoid chattering. As the gain varies, the proportion of self-delayed feedback is also adapted to improve the tracking performance and minimize introducing undesired oscillations into the system.

Each follower agent has a FLC system that receives $\boldsymbol{\eta}_i$, $i \in \mathbb{N}$, as an input, which is calculated using an exponential moving average function

$$\boldsymbol{\eta}_i(t) = \boldsymbol{\theta}^{-1}[\frac{1}{\varepsilon(t)}\hat{\boldsymbol{s}}_i \circ \dot{\hat{\boldsymbol{s}}}_i + (1 - \frac{1}{\varepsilon(t)})\boldsymbol{\eta}_i(t - T_s)], \tag{5.35}$$

where $\boldsymbol{\theta}$ is a normalization factor diagonal matrix used to scale the input in the interval $[-1, 1]$, T_s is the sampling time, $\varepsilon(T_s) = 1$, $\varepsilon(t) = \phi\varepsilon(t - T_s) + 1$, ϕ is a positive scalar forgetting factor, and $\circ$ represents the element-wise multiplication. The sliding surface from the previous time step is calculated as

$$\hat{\boldsymbol{s}}_i = \hat{\boldsymbol{e}}_i + \beta(\dot{\hat{\boldsymbol{e}}}_i)^{\alpha},$$

where

$$\hat{\boldsymbol{e}}_i = \frac{1}{\mu_i}\sum_{j=1}^{\mathbb{N}} a_{ij}[\boldsymbol{x}_i(t - T_s) - \boldsymbol{x}_j(t - T_s)] + b_i[\boldsymbol{x}_i(t - T_s) - \boldsymbol{x}_0(t - T_s)],$$

$$\dot{\boldsymbol{e}}_i = \frac{1}{\mu_i}\sum_{j=1}^{\mathbb{N}} a_{ij}[\dot{\boldsymbol{x}}_i(t-T_s) - \dot{\boldsymbol{x}}_j(t-T_s)] + b_i[\dot{\boldsymbol{x}}_i(t-T_s) - \dot{\boldsymbol{x}}_0(t-T_s)],$$

and

$$\mu_i = \sum_{j=1}^{\mathbb{N}} a_{ij} + \zeta b_i,$$

where μ_i is a neighbour scaling factor used to adjust the errors of each agent onto the same scale and ζ is the leader connection factor that is determined by user experience. A saturation function is applied to restrict the input to an interval $[-1, 1]$. The exponential moving average is applied to retain past error information and was found to improve the FLC performance. The outputs of the fuzzy controller, $\Delta\boldsymbol{\kappa}_i$ and $\Delta\boldsymbol{\varpi}_{1_i}$, are restricted to an interval $[0, 1]$ and are used to tune the parameters by

$$\boldsymbol{\kappa}_i = G_{\boldsymbol{\kappa}} \int_0^t \Delta\boldsymbol{\kappa}_i dt + \boldsymbol{\kappa}_{\min}, \tag{5.36}$$

$$\boldsymbol{\varpi}_{1_i} = G_{\boldsymbol{\varpi}_1} \int_0^t \Delta\boldsymbol{\varpi}_{1_i} dt + \boldsymbol{\varpi}_{1_{\min}}, \tag{5.37}$$

where $G_* = G((*)_{\max} - (*)_{\min})$, G is a tuning parameter defined by the user, and $(*)_{\max}$, $(*)_{\min}$ are designed as the upper and lower limits of the respective controller parameters. Note that $(*)_{min} > 0$ to guarantee the stability of the controller. Saturation functions with upper and lower limits of $(*)_{\max}$ and $(*)_{\min}$ are applied to $\boldsymbol{\kappa}_i$ and $\boldsymbol{\varpi}_{1_i}$. The initial conditions of the integrals are set as G^{-1}.

The input and output of the fuzzy controller are designed with five membership functions, resulting in a total of ten fuzzy rules. Fig. 5.1 shows the membership functions for the inputs and outputs. The significant reduction in fuzzy rules, compared to the FLC method in Chapter 4, is a result of reducing the number of inputs, from two to one, and outputs, from three to two. The reduction in the number of fuzzy rules considerable reduces the computational time and therefore allows independent FLC system to be applied to each of the state dimensions. Table 5.1 shows the rule bases for $\Delta\boldsymbol{\kappa}_i$ and $\Delta\boldsymbol{\varpi}_{1_i}$. The defuzzification is performed using the center of area method.

Remark 10. *Provided that the operating range of $\boldsymbol{\kappa}_i$ and $\boldsymbol{\varpi}_1$ defined in the FLC controller in (5.36) and (5.37) satisfy the inequalities in (5.26) and (5.29), then the*

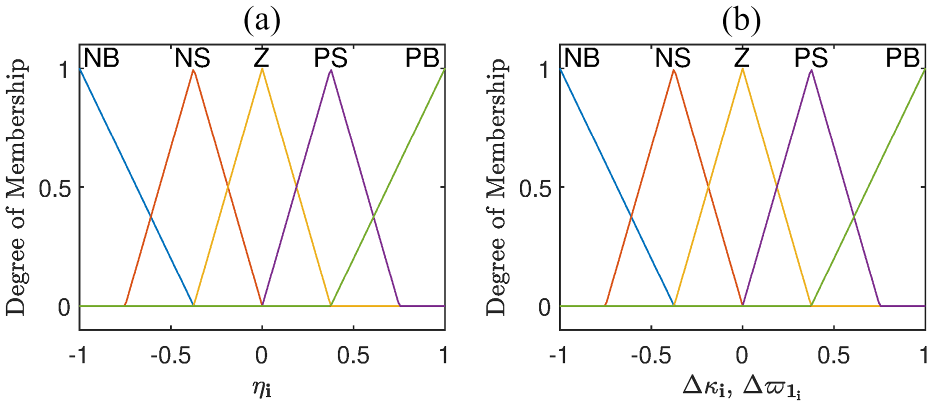

Figure 5.1: (a) Input membership functions and (b) output membership functions for NTSMC+FLC

Table 5.1: Rule base for $\Delta\boldsymbol{\kappa}_i$ and $\Delta\boldsymbol{\varpi}_{1_i}$

$\boldsymbol{\eta}_i$	NB	NS	Z	PS	PB
$\Delta\boldsymbol{\kappa}_i$	NB	NS	Z	PS	PB
$\Delta\boldsymbol{\varpi}_{1_i}$	NB	NS	Z	PS	PB

stability of the system is guaranteed.

5.3 Simulations and Experiments

This section describes the simulation and experimental parameters for synchronizing a networked group of EL Phantom Omni manipulators, analyzes the improvement in tracking performance for the simulations, and validates the proposed control policy in an experimental setting.

5.3.1 Simulation Model

The focus of the following simulations are on a group of five agents with the leader-follower communication topology in Fig. 5.2 is considered in the following simulations.

The communication graph is represented as

$$\mathcal{A} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}, \qquad \mathcal{B} = \begin{bmatrix} 1 & 1 & 0 & 0 \end{bmatrix}^T. \tag{5.38}$$

The dynamics of the MASs are as described in Section 3.5. The leader is assigned a circular trajectory in the x-y plane that is experimentally recorded with a Phantom Omni device, and is approximately

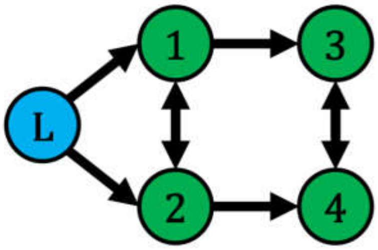

Figure 5.2: Leader-follower communication topology for EL simulations

$$\boldsymbol{x}_0(t) = \begin{bmatrix} -0.0185\cos(\frac{\pi}{14}t) - 0.0115 \\ -0.025\sin(\frac{\pi}{14}t) \\ -0.06 \end{bmatrix} m. \tag{5.39}$$

The velocity and acceleration of the leader are also experimentally determined as $\dot{\boldsymbol{x}}_{\boldsymbol{0}} = \frac{\boldsymbol{dx_0}}{\boldsymbol{dt}}\ m/s$ and $\ddot{\boldsymbol{x}}_{\boldsymbol{0}} = \frac{\boldsymbol{d\dot{x}_0}}{\boldsymbol{dt}}\ m/s^2$. Each communication link is subjected to a time-varying delay of $T_{ij}(t) = 0.1\ s$ and $\delta_{ij} = 0.025\ s$. At $t = 15\ s$, Agent 4 experiences a change in delay to $T_{ij}(t) = 0.4\ s$ and $\delta_{ij} = 0.2\ s$ to simulate a sudden drop in the network quality. To validate the robustness of the controller, a disturbance, $\boldsymbol{d}_2^{\tau}$, is applied to Agent 2 and is defined as

$$\boldsymbol{d}_2^{\tau} = J^T \boldsymbol{d}_2^u\ Nm,$$

where

$$\boldsymbol{d}_2^u = \begin{bmatrix} 3e^{(\frac{-(t-7)^2}{0.32})} & 0 & 0 \end{bmatrix}^T N.$$

Note that the disturbance is designed in task space as $\boldsymbol{d}_2^u$ and applied in joint space as $\boldsymbol{d}_2^\tau$. The disturbances in both task and joint space are plotted in Fig. 5.3.

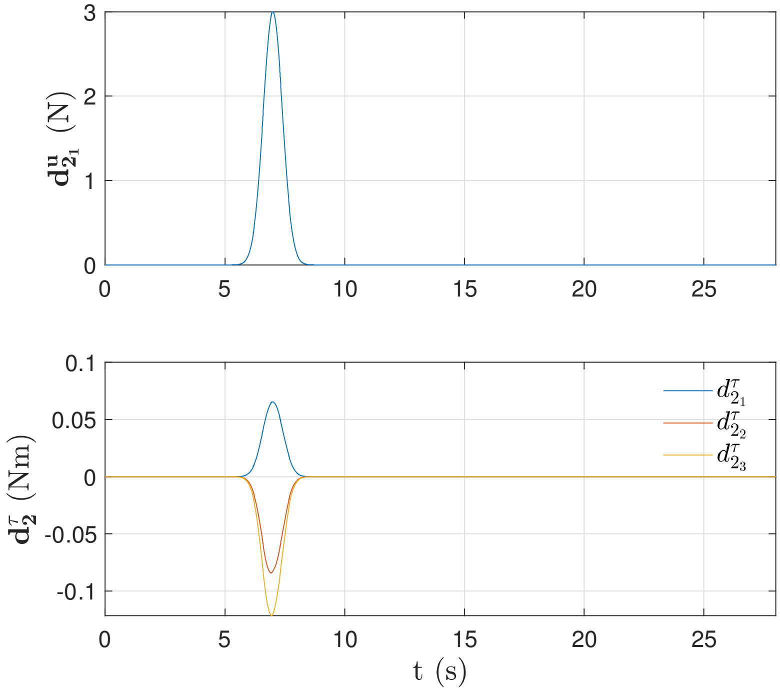

Figure 5.3: The disturbances in task and joint space, $\boldsymbol{d}_2^u$ and $\boldsymbol{d}_2^\tau$, applied to Agent 2, for simulation with $\kappa = 30$ and $\varpi_1 = 1.00$

The initial velocities of all agents are zero and the initial positions of the agents are (units: m)

$$\boldsymbol{x}_0 = \begin{bmatrix} -0.03 \\ 0 \\ -0.06 \end{bmatrix}, \boldsymbol{x}_1 = \begin{bmatrix} 0.03 \\ -0.01 \\ -0.065 \end{bmatrix}, \boldsymbol{x}_2 = \begin{bmatrix} -0.03 \\ 0.01 \\ -0.065 \end{bmatrix}, \boldsymbol{x}_3 = \begin{bmatrix} 0.02 \\ -0.02 \\ -0.065 \end{bmatrix}, \boldsymbol{x}_4 = \begin{bmatrix} -0.02 \\ 0.02 \\ -0.065 \end{bmatrix}.$$

5.3.2 Simulation Results

The simulation outlined in the previous section is run with various constant values of $\boldsymbol{\kappa}_i = \boldsymbol{\kappa}$ and $\boldsymbol{\varpi}_{1_i} = \boldsymbol{\varpi}_1$, $i \in \{1, \dots, N\}$. The RMSE and maximum error for different constant values of $\boldsymbol{\kappa}$, depicted by each color set, and $\boldsymbol{\varpi}_1$, shown on the x-axis, are presented in Figs. 5.4 and 5.5 in the range of $\boldsymbol{\varpi}_1 \in \{0.75, \dots, 1\}$. It is observed that the trends for the RMSE and maximum error are similar in the ranges presented.

Note that the RMSE and maximum error continue to increase exponentially lower values of $\boldsymbol{\varpi}_1$.

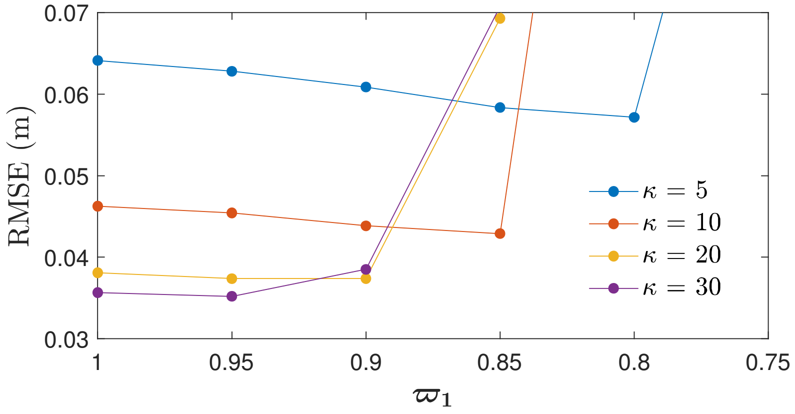

Figure 5.4: Simulation RMSE with various constant values of $\boldsymbol{\kappa}$ and $\boldsymbol{\varpi}_1$ for the topology in Fig. 5.2

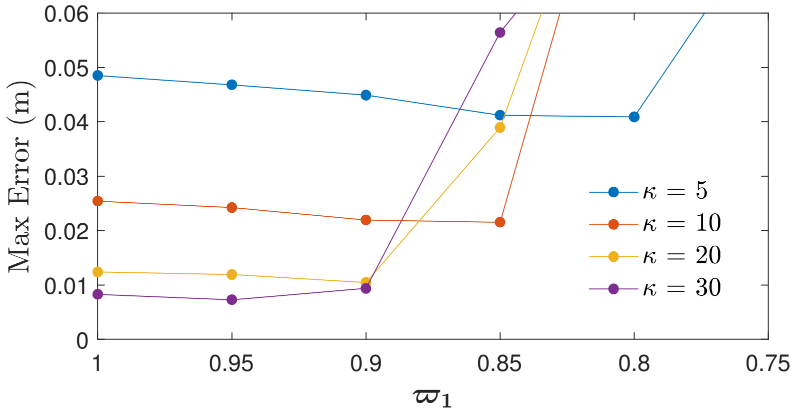

Figure 5.5: Simulation maximum error with various constant values of $\boldsymbol{\kappa}$ and $\boldsymbol{\varpi}_1$ for the topology in Fig. 5.2

Simulations of a fully connected communication topology for one leader and four followers were conducted and the results are presented in Figs. 5.6 and 5.7. Compared to Figs. 5.4 and 5.5, the RMSE and maximum error follow a similar trend, but the measure of self-delay can be raised further. This allows for a higher potential of improvement from the proposed control method for communication topologies with more connections as there is a great range of $\boldsymbol{\varpi}_1$ to operate within. It is also noted that more communication links does not necessarily result in a lower RMSE. Since each agent is receiving information from all other agents, they are driven towards the average state of all agents rather then to the leader agent state. Additionally, simulations of a smaller fully connected communication topology for one leader and two followers were conducted and the results are presented in Figs. 5.8 and 5.9. It is observed that the case with fewer agents follows the same trend as the two previous cases but the measure of self-delay can not be as high as the fully connected case. This indicates that for systems with fewer agents, the potential of improvement from the proposed control method is lower.

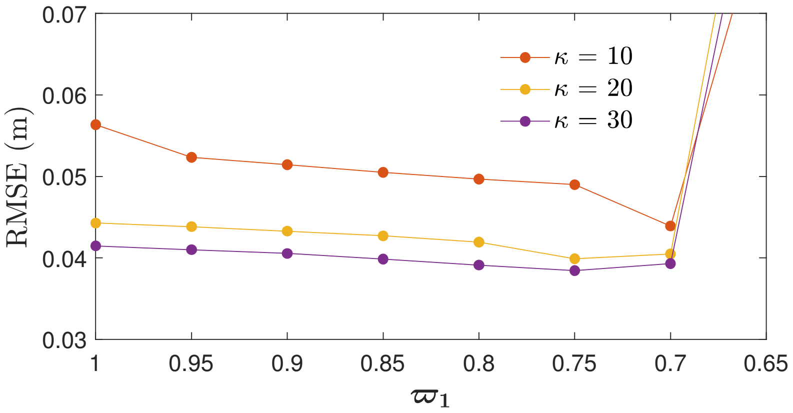

Figure 5.6: Simulation RMSE with various constant values of $\boldsymbol{\kappa}$ and $\boldsymbol{\varpi}_1$ for a fully connected topology of five agents

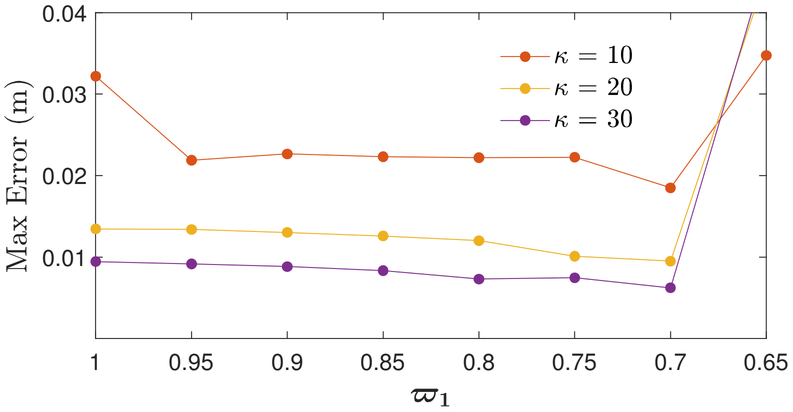

Figure 5.7: Simulation maximum error with various constant values of $\boldsymbol{\kappa}$ and $\boldsymbol{\varpi}_1$ for a fully connected topology of five agents

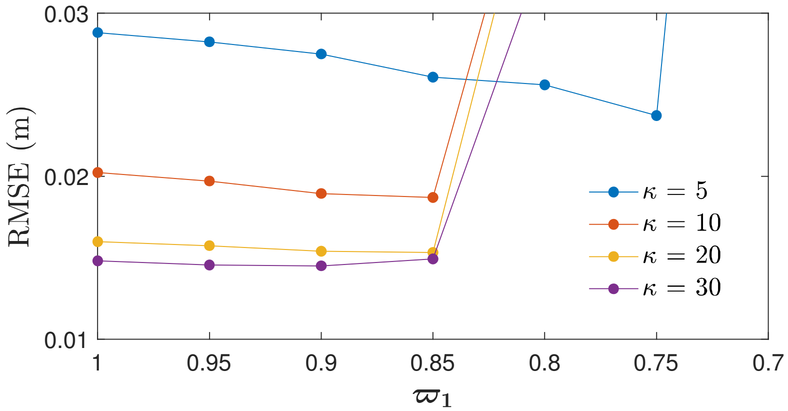

Figure 5.8: Simulation RMSE with various constant values of $\boldsymbol{\kappa}$ and $\boldsymbol{\varpi}_1$ for a fully connected topology of three agents

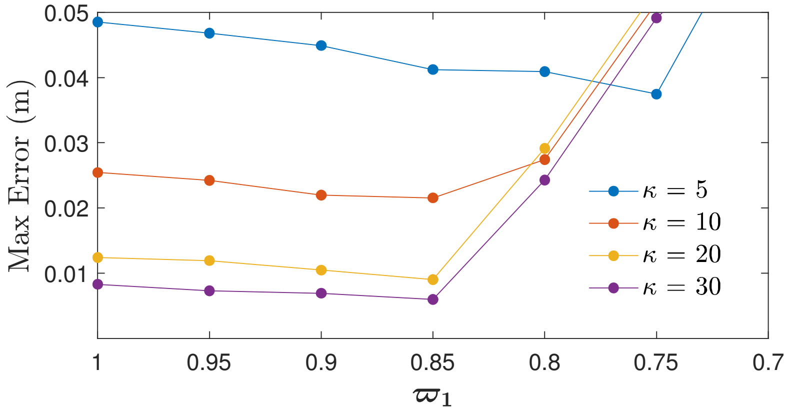

Figure 5.9: Simulation maximum error with various constant values of $\boldsymbol{\kappa}$ and $\boldsymbol{\varpi}_1$ for a fully connected topology of three agents

From Figs. 5.4 and 5.5, two observations can be made. The first is that as $\boldsymbol{\varpi}_1$ decreases and more self-delayed feedback is introduced into the system, the error decreases to an extent before it begins to increase. The second is that the extent that $\boldsymbol{\varpi}_1$ can be lowered, decreases as $\boldsymbol{\kappa}$ is increased. These two observations indicate that self-delayed feedback can have more influence on the performance at lower values of $\boldsymbol{\kappa}$ and that there exists varying optimal values of $\boldsymbol{\varpi}_1$ for different values of $\boldsymbol{\kappa}$. Based on these observations, the parameters were tuned for the FLC.

The following results are for the communication topology in Fig. 5.2. Figs. 5.10 - 5.12 show the positions of the simulated manipulators in task space for high gain values of $\kappa = 30$ and various constant cases of $\boldsymbol{\varpi}_1 = 1.00$, $\boldsymbol{\varpi}_1 = 0.95$, and $\boldsymbol{\varpi}_1 = 0.90$. It is observed that introducing small proportions of self-delay into the system with high values of $\boldsymbol{\kappa}$ introduces significant chattering in all axes.

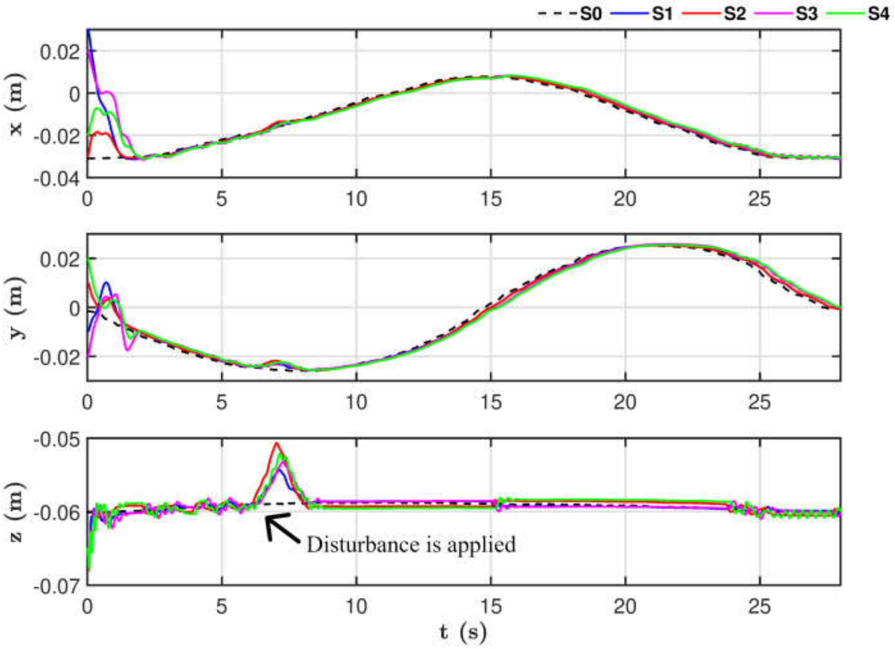

Figure 5.10: Simulation agents' end effector positions for $\boldsymbol{\kappa} = 30$ and $\boldsymbol{\varpi}_1 = 1.00$ with NTSMC

Figs. 5.13 and 5.14 show the positions of the simulated manipulators in task space for low gain values of $\kappa = 5$ and constant cases of $\boldsymbol{\varpi}_1 = 1.00$ and $\boldsymbol{\varpi}_1 = 0.80$. It is observed that, compared to the high gain cases in Figs. 5.10 - 5.12, the followers do not track the leader as closely and the disturbance causes a much higher deviation of the follower's positions from the leader's position. Another observation in comparing Figs. 5.12 and 5.14 is that for lower values of $\boldsymbol{\kappa}$, introducing self-delay into the system does not introduce oscillations as quickly as in the cases of high values of $\boldsymbol{\kappa}$.

The case for the NTSMC+FLC is plotted in Fig. 5.15 and the time-varying parameters, $\boldsymbol{\kappa}_i$ and $\boldsymbol{\varpi}_{1_i}$ are shown in Figs. 5.16 and 5.17. The controller parameters are presented in Table 5.2. From Fig. 5.15, the NTSMC+FLC method has fewer oscillations with lower amplitudes than the case in Fig. 5.11. The total RMSE and maximum error values for the three cases are shown in Table 5.3 which shows that the NTSMC+FLC method achieves the reductions in the maximum error due to the

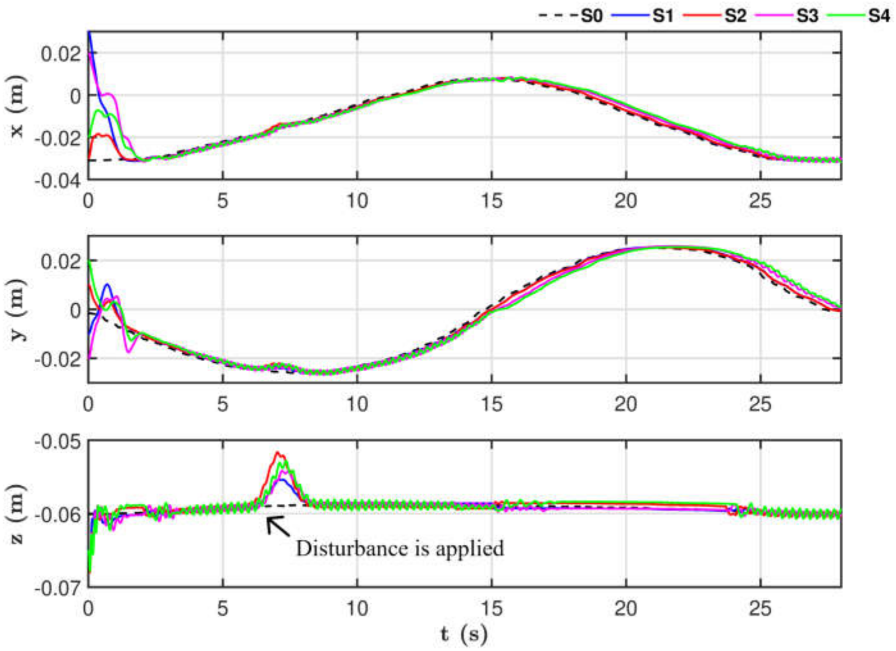

Figure 5.11: Simulation agents' end effector positions for $\boldsymbol{\kappa} = 30$ and $\boldsymbol{\varpi}_1 = 0.95$ with NTSMC

disturbance applied at $t = 6\ s$, and an RMSE between the two constant cases of $\boldsymbol{\varpi}_1 = 1.00$ and $\boldsymbol{\varpi}_1 = 0.95$. This indicates that the NTSMC+FLC method grants a trade-off between improving the tracking performance and reducing the level of oscillations, while minimizing the maximum error due to disturbances.

Remark 11. *The FLC parameters are determined by user experience and by tuning the simulation. For example, θ is approximated as the maximum value of 5.35 for the case with constant parameters during the steady state period. The parameters G, ϕ, and ζ are determined by tuning each individually and observing the effect on the performance of the MAS. Based on the observations from the results in Fig. 5.4, the maximum and minimum limits of $\boldsymbol{\kappa}$ and $\boldsymbol{\varpi_1}$ parameters can be established.*

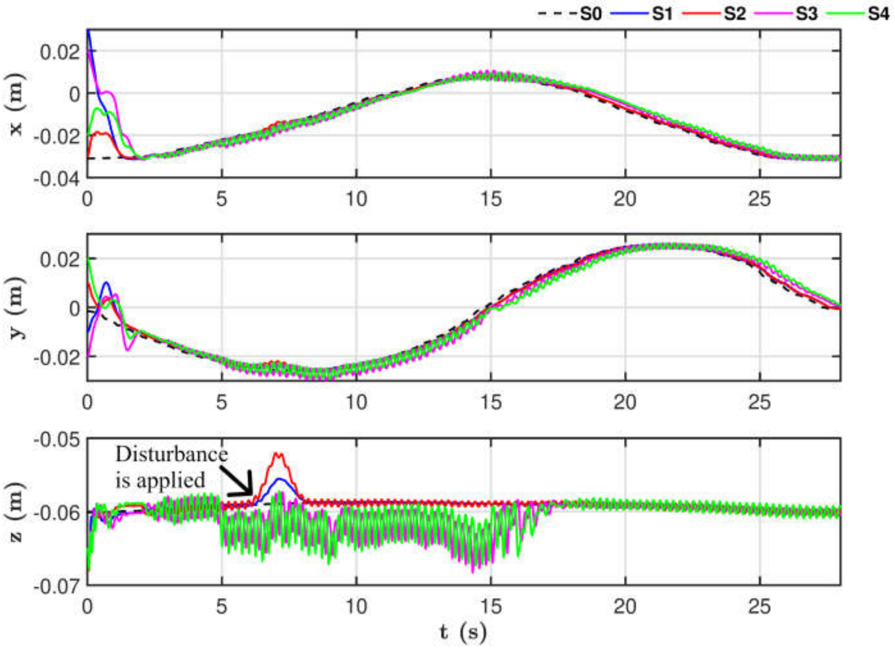

Figure 5.12: Simulation agents' end effector positions for $\boldsymbol{\kappa} = 30$ and $\boldsymbol{\varpi}_1 = 0.90$ with NTSMC

Table 5.2: FLC parameters for NTSMC+FLC simulations

$\boldsymbol{\theta}$	$2 \times 10^{-6}\mathbf{I_3}$	G	1
$\boldsymbol{\kappa}_{\max}$	30	$\boldsymbol{\varpi}_{1_{\max}}$	1
$\boldsymbol{\kappa}_{\min}$	5	$\boldsymbol{\varpi}_{1_{\min}}$	0.9
α	0.98	ζ	4

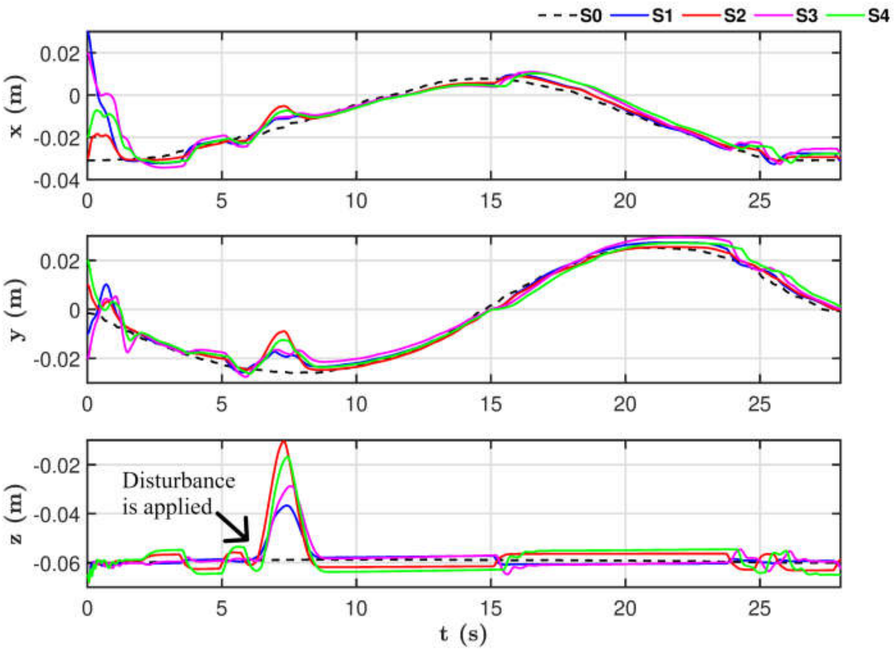

Figure 5.13: Simulation agents' end effector positions for $\boldsymbol{\kappa} = 5$ and $\boldsymbol{\varpi}_1 = 1.00$ with NTSMC

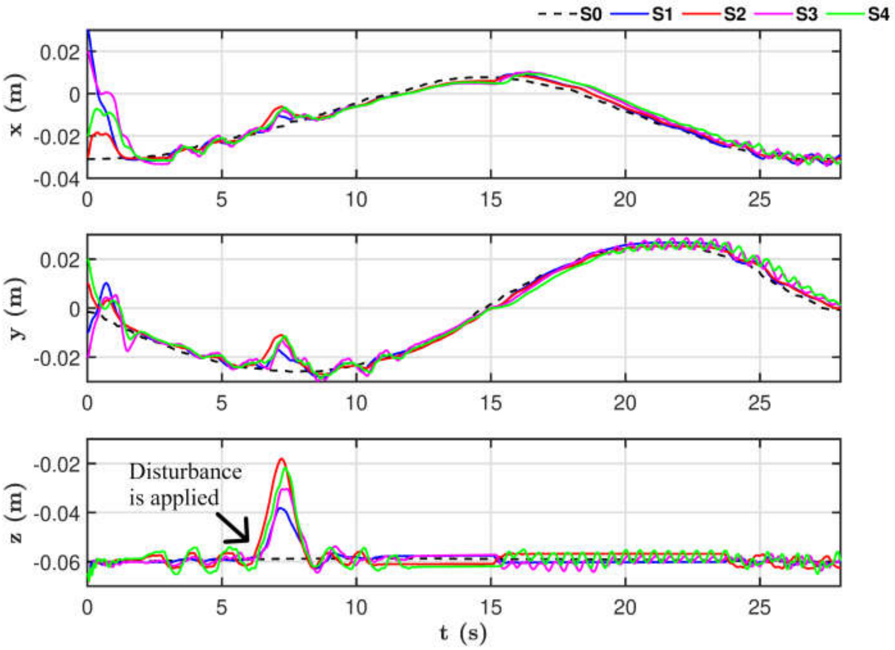

Figure 5.14: Simulation agents' end effector positions for $\boldsymbol{\kappa} = 5$ and $\boldsymbol{\varpi}_1 = 0.80$ with NTSMC

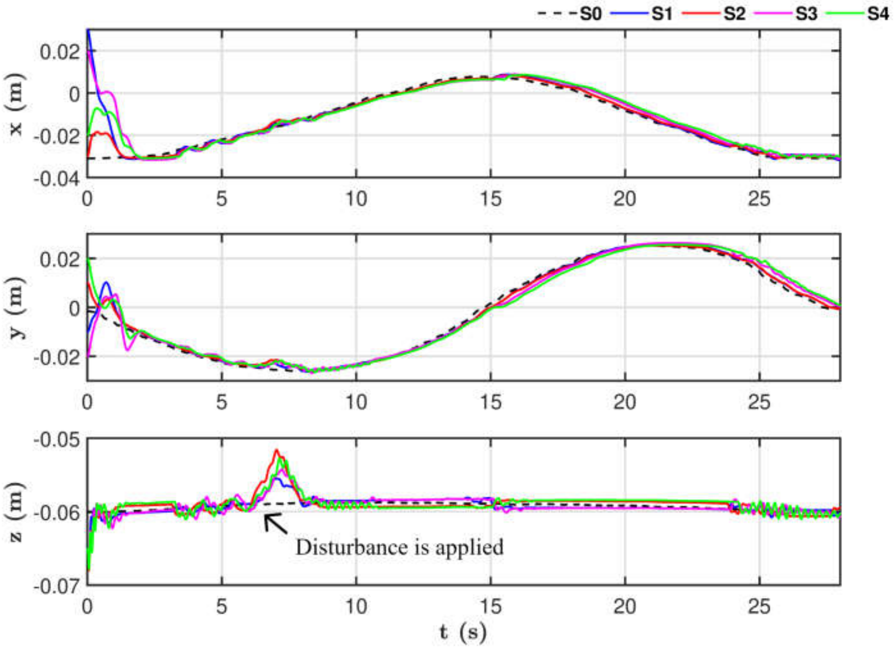

Figure 5.15: Simulation agents' end effector positions for $\boldsymbol{\kappa}_i$ and $\boldsymbol{\varpi}_{1_i}$ with NTSMC+FLC

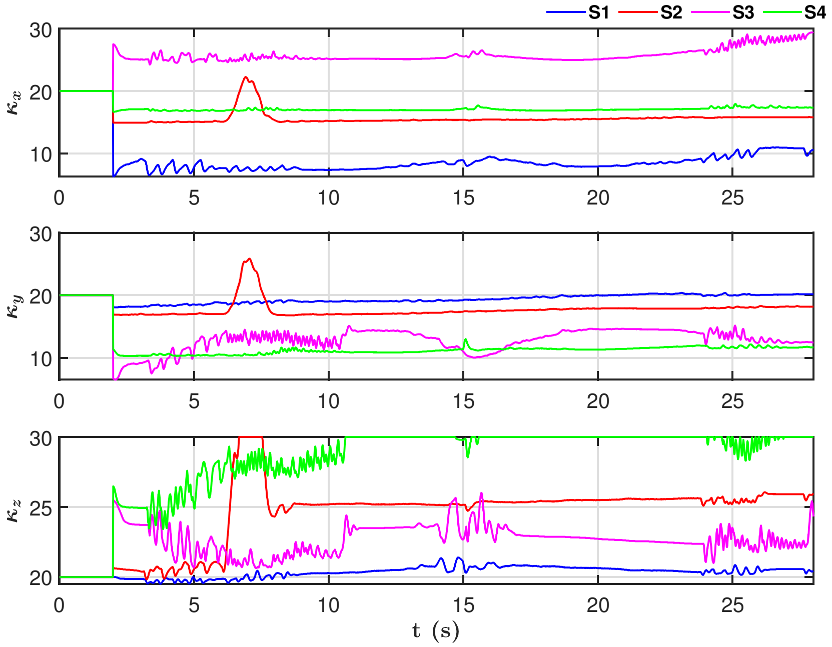

Figure 5.16: Simulation time-varying $\boldsymbol{\kappa}_i$ tuned with NTSMC+FLC

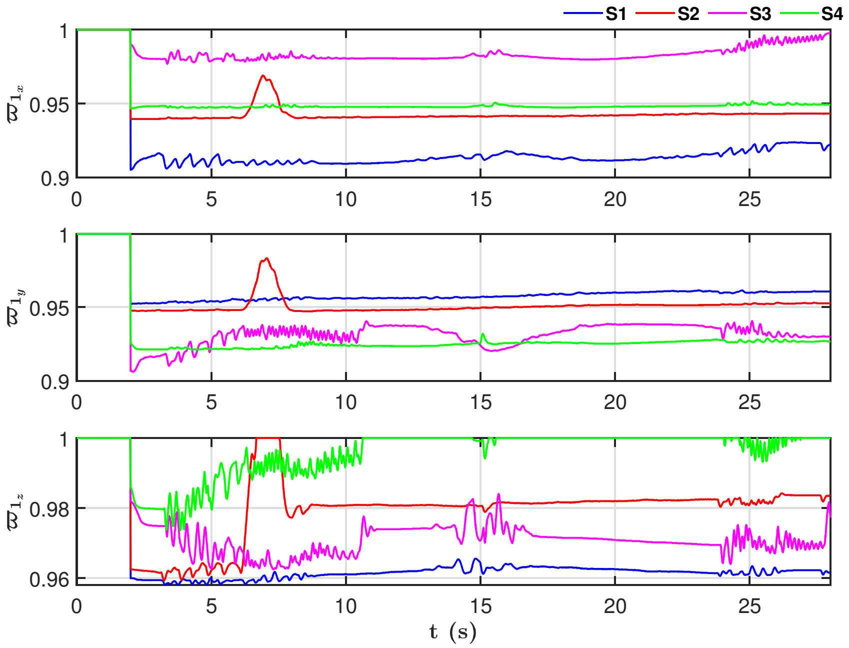

Figure 5.17: Simulation time-varying $\boldsymbol{\varpi}_{1_i}$ tuned with NTSMC+FLC

Table 5.3: Simulation performance metrics

Case	RMSE (m)	Max Error (m)
$\boldsymbol{\kappa} = 5$, $\boldsymbol{\varpi}_1 = 1$	0.0642	0.049
$\boldsymbol{\kappa} = 5$, $\boldsymbol{\varpi}_1 = 0.8$	0.0572	0.041
$\boldsymbol{\kappa} = 30$, $\boldsymbol{\varpi}_1 = 1$	0.0357	0.0083
$\boldsymbol{\kappa} = 30$, $\boldsymbol{\varpi}_1 = 0.95$	0.0351	0.0073
$\boldsymbol{\kappa} = 30$, $\boldsymbol{\varpi}_1 = 0.9$	0.0385	0.0094
FLC	0.0354	0.0073

5.3.3 Experimental Setup

The NTSMC+FLC controller is applied to two manipulators with a virtual leader, considering the availability of the hardware in the lab. The experimental setup of two Phantom Omni devices and two computers is shown in Fig. 5.18. The connection between the computers and the devices are established with Matlab/Simulink and Quanser QUARC 2.6. The connection between the two computers is established with UDP with a frequency of 1000 Hz and a wired Ethernet local area network (LAN). The time delays are measured of the wired connection are measured to be 1.25 ± 0.25 ms. A wired Ethernet connections is used instead of a wireless connection to minimize the existence of other communication constraints, such as packet loss and other interferences, to avoid damage to the devices. In practical cases, the communication between agents may be established through a wireless LAN and the time delays would be higher [22]. Therefore, in the following experiment, additional time delays are applied in the controller to impose communication delays as if the agents were operating a wireless network.

The communication topology is shown in Fig. 5.19. An additional delay of $T_{ij}(t) = 0.05$ s and $\delta_{ij} = 0.025$ s was added to the each agent. An external disturbance is physically applied to Agent 1 at $t = 6$ s in Trial 1 and $t = 10$ s in Trial 2.

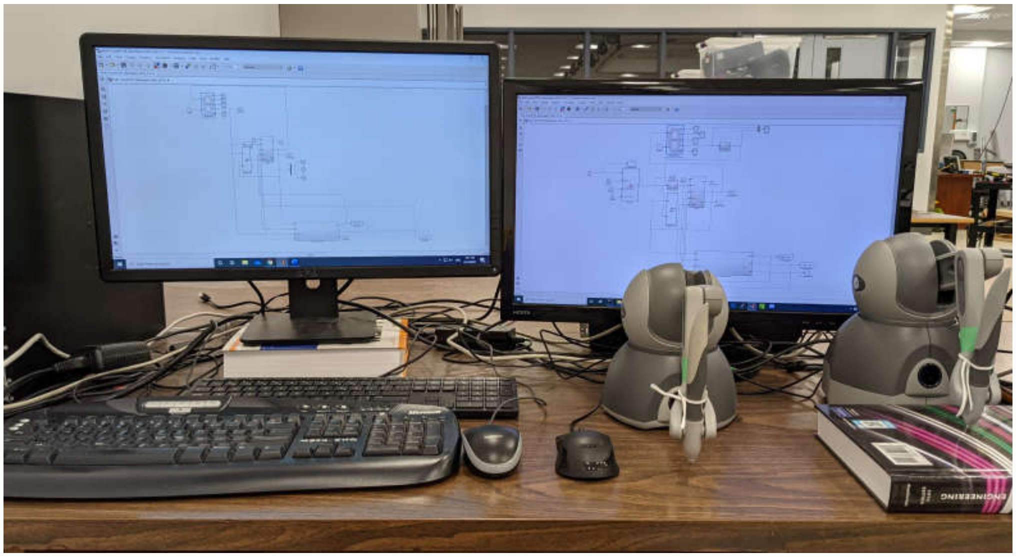

Figure 5.18: Experimental setup of two Phantom Omni devices

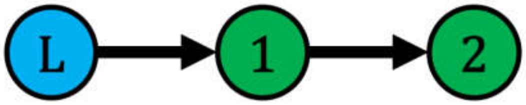

Figure 5.19: Leader-follower communication topology for multiple Phantom Omni systems

5.3.4 Experimental Results

Two trials of applying the NTSMC+FLC method are conducted with the experimental setup. The results of the first trial are presented in Figs. 5.20 - 5.22. The RMSE for Trial 1 is 0.0157 m. In Fig. 5.20, it can be observed that there is significant chatter after the disturbance is applied. It is also observed in Figs. 5.21 and 5.22, that the parameters κ_y and ϖ_{1_y} reach their maximum values and remain constant throughout the experiment.

The controller parameters are further tuned for the second trial and the results are presented in Figs. 5.23 - 5.25. The RMSE for Trial 2 is 0.0130 m. In Fig. 5.23, it can be observed that, compared to Trial 1, the chatter is significantly reduced after the disturbance is applied. The parameters in Figs. 5.24 and 5.25 follow similar trends as in Trial 1 though there is more variation in the y-axis parameters.

The variation in performance and response to the disturbance between the two trials could be attributed to several factors. The possible factors include the time at which the disturbances were applied, the magnitude of the disturbances, the tuning parameters, or differences in the hardware's friction. Additional tuning of the control parameters may result in a further improvement in performance.

In each trial, both of the follower agents recover and continue to synchronize with the leader. The results in Figs. 5.20 and 5.23 show a successful application of the NTSMC+FLC controller to a group of real manipulators and demonstrates the robustness of the controller.

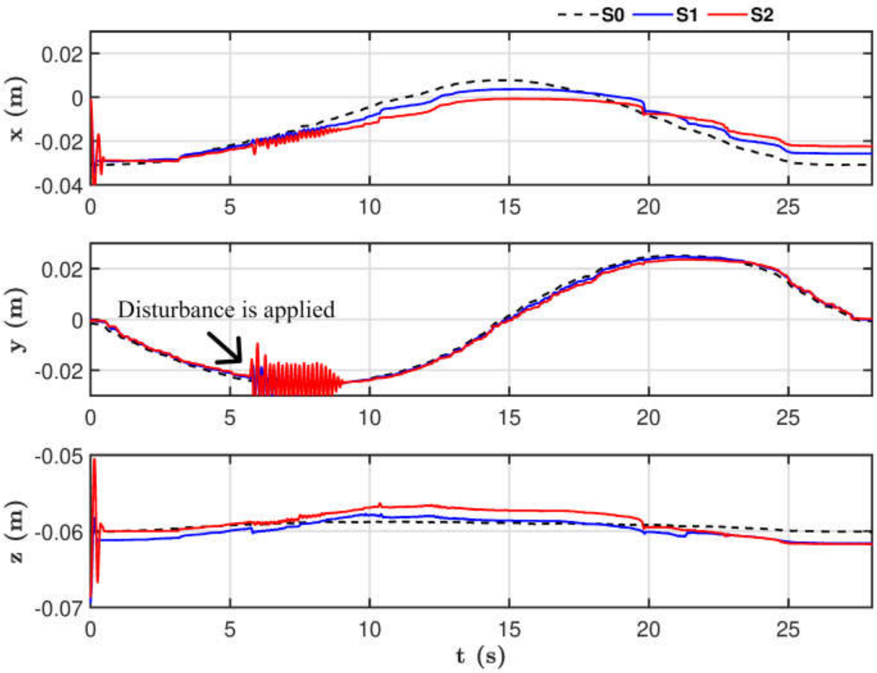

Figure 5.20: Experimental agents' end effector positions for Trial 1

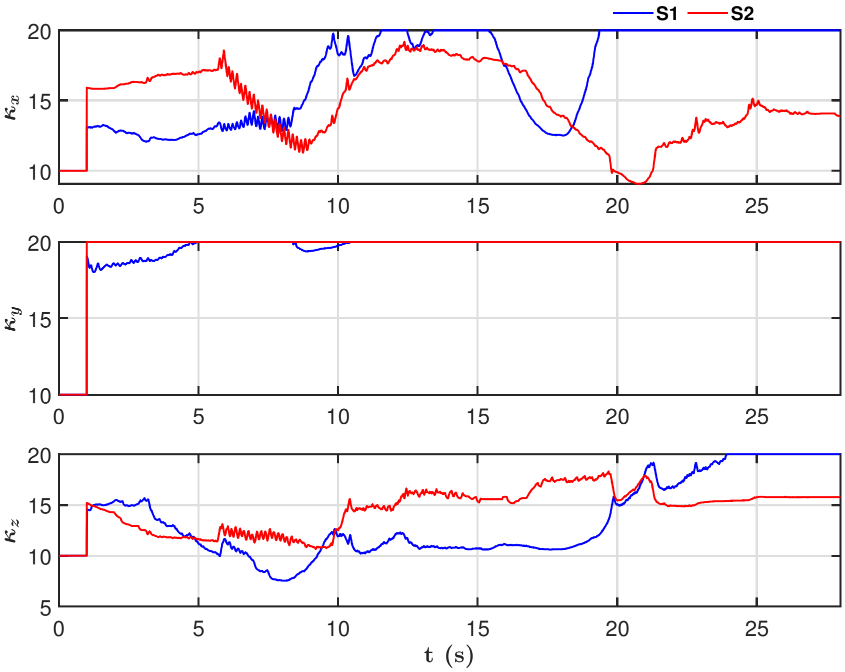

Figure 5.21: Experimental time-varying $\boldsymbol{\kappa}_i$ for Trial 1

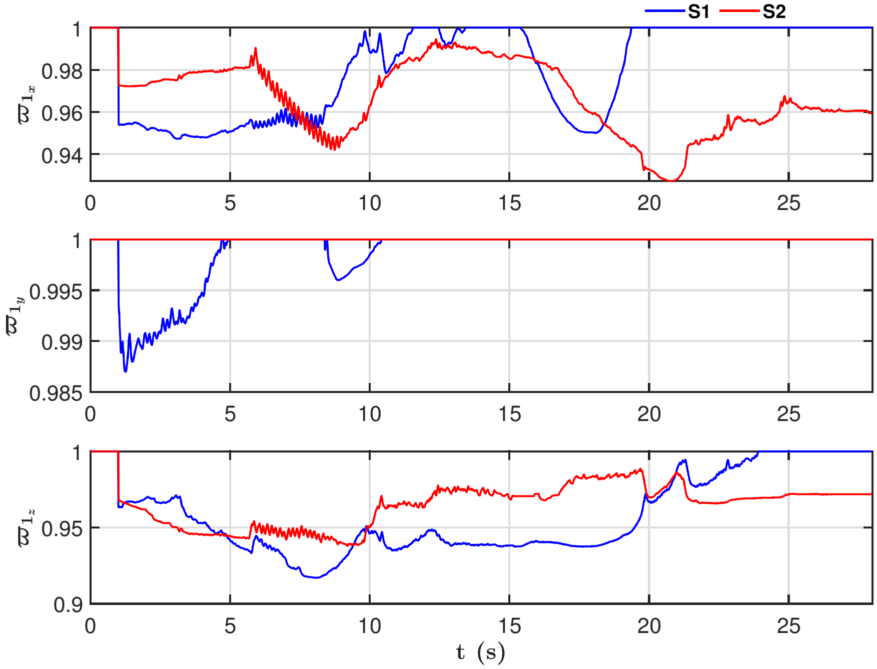

Figure 5.22: Experimental time-varying ϖ_{1_i} for Trial 1

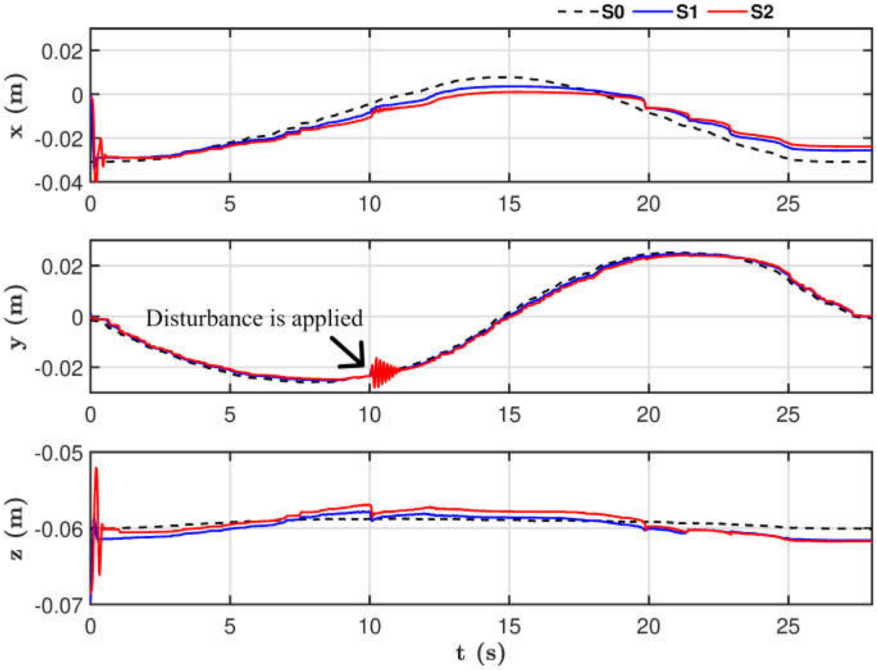

Figure 5.23: Experimental agents' end effector positions for Trial 2

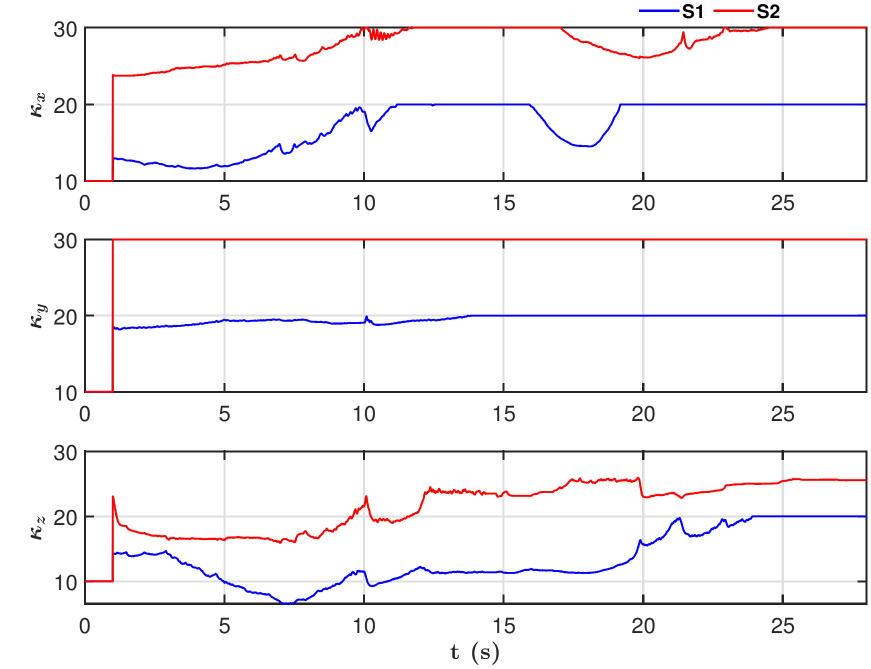

Figure 5.24: Experimental time-varying $\boldsymbol{\kappa}_i$ for Trial 2

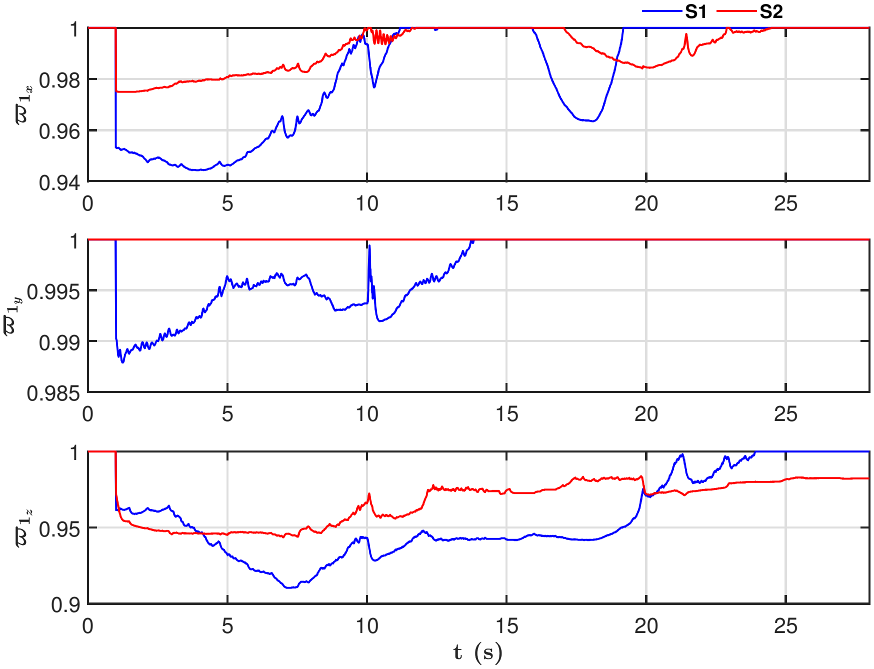

Figure 5.25: Experimental time-varying $\boldsymbol{\varpi}_{1_i}$ for Trial 2

5.4 Summary

The FLC method from the previous chapter has been improved and applied to NTSMC for nonlinear MASs. Simulations of a group of 3-DOF manipulators show that increasing the proportion of self-delay in the controller feedback improves the overall synchronization tracking performance and reduces the impact of external disturbances on the performance. It is found that communication networks with more agents and more connections allow for a higher proportion of self-delay to be introduced into the controller. Simulations show that the proposed NTSMC+FLC method grants a trade-off between improving the performance and reducing the level of undesirable oscillations, while improving the response to disturbances. Experiments with Phantom Omni manipulators have been successfully conducted to validate the proposed NTSMC+FLC controller.

Chapter 6

Conclusions

This chapter summarizes the work in this thesis and suggests some research areas that extend the concept of applying FLC to the control of linear and EL networked robotic MASs.

6.1 Conclusions

A novel control method that applies the concept of mixed-type feedback and tunes the control parameters online using FLC has been applied to linear and nonlinear EL MASs to address the synchronization problem. In the case of linear MASs, a method of deriving a feasible, stable range of gains using LMI and Lyapunov methodology is presented. A basic approach to using FLC to tune the parameters online is developed and simulations are conducted, demonstrating an improvement in overall performance of the system. For the EL MASs, a NTSMC policy is proposed and the stability is proven for a MAS with time-varying control parameters. The FLC system is improved upon and both simulations and experiments of groups of 3-DOF manipulators are conducted to demonstrate the performance improvement and to validate the controller.

6.2 Future Work

There are several potential topics that could be pursued based on this work. The first is that a systematic method of determining the FLC parameters could be developed. Currently, the parameters are tuned manually by the user in the simulated environment. An automated or more systematic approach would increase the feasibility of using the proposed control policy for tasks in a practical setting. A mathematical solution considering the system dynamics, communication topology, and communication delays could be established and used to suggest the optimal or feasible FLC parameters.

The second area for future work is to apply the proposed control policy to more complex experimental setups. The experimental environment in this work was limited to two physical manipulators due to limited available equipment. Applying the proposed control method to MASs with more agents, sets of heterogeneous agents, or more complex communication toplogies would further validate the controller.

The third area for future work is to investigate methods of predicting the estimated self-delay value online. In this work, the self-delay value is assumed to be known and defined as the average of the time-varying delays. In a real-world setting, the average of the time-varying delays may be unknown, may change during the task, or may be different values for different communication links. Therefore, developing an online method of estimating the average communication between agents would be a crucial step when using the control method proposed in this work.

Finally, the forth potential research topic is applying the proposed control policy to MASs that have time-varying communication delays that are correlated in time. In this work, the time-varying delays are considered as uncorrelated in time. A representation of the time-delays by a model such as red noise may be a more accurate representation of communication networks and may introduce more complexities into the control of MASs. Considering that fuzzy logic is well equipped for nonlinear systems, the control method proposed in this work may have more significant advantages over other control methods in controlling MASs with communication delays represented by more complex models.

Bibliography

[1] FutureAtlas, "Military robot," 2016. Accessed on: June 13, 2021. [Online]. Available: `https://www.flickr.com/photos/87913776@N00/27618450381`.

[2] ICAPlants, "Ribituzed float glass unloading," 2008. Accessed on: June 13, 2021. [Online]. Available: `https://commons.wikimedia.org/wiki/File:Float_Glass_Unloading.jpg`.

[3] S. Forbrigger, "Prediction-based haptic interfaces to improve transparency for complex virtual environments," Master's thesis, Dalhousie University, Nova Scotia, Canada, 2017.

[4] S. Grayson, "Search & rescue using multi-robot systems," *School of Computer Science and Informatics, University College Dublin*, 2014.

[5] Y. Liu, *Semi-Autonomous Control of Multi-Robot Teams in Urban Search and Rescue Applications.* PhD thesis, Toronto University, Ontario, Canada, 2019.

[6] C. Y. Wong, G. Seet, and S. K. Sim, "Multiple-robot systems for usar: key design attributes and deployment issues," *International Journal of Advanced Robotic Systems*, vol. 8, no. 1, p. 12, 2011.

[7] J. A. Marvel, R. Bostelman, and J. Falco, "Multi-robot assembly strategies and metrics," *ACM Computing Surveys (CSUR)*, vol. 51, no. 1, pp. 1–32, 2018.

[8] M. Defoort, A. Polyakov, G. Demesure, M. Djemai, and K. Veluvolu, "Leader-follower fixed-time consensus for multi-agent systems with unknown non-linear inherent dynamics," *IET Control Theory & Applications*, vol. 9, no. 14, pp. 2165–2170, 2015.

[9] G. He and H. Li, "Distributed control for multirobot systems with collision-free motion coordination," in *2017 10th International Symposium on Computational Intelligence and Design (ISCID)*, vol. 2, pp. 72–76, IEEE, 2017.

[10] N. S. Morozova, "Formation control and obstacle avoidance for multi-agent systems with dynamic topology," in *2015 International Conference" Stability and Control Processes" in Memory of VI Zubov (SCP)*, pp. 580–583, IEEE, 2015.

[11] Z. Li, G. Wen, Z. Duan, and W. Ren, "Designing fully distributed consensus protocols for linear multi-agent systems with directed graphs," *IEEE Transactions on Automatic Control*, vol. 60, no. 4, pp. 1152–1157, 2014.

[12] H. Du, Y. He, and Y. Cheng, "Finite-time synchronization of a class of second-order nonlinear multi-agent systems using output feedback control," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 61, no. 6, pp. 1778–1788, 2014.

[13] B. Tian, H. Lu, Z. Zuo, and W. Yang, "Fixed-time leader–follower output feedback consensus for second-order multiagent systems," *IEEE transactions on cybernetics*, vol. 49, no. 4, pp. 1545–1550, 2018.

[14] J. R. Klotz, Z. Kan, J. M. Shea, E. L. Pasiliao, and W. E. Dixon, "Asymptotic synchronization of a leader-follower network of uncertain euler-lagrange systems," *IEEE Transactions on Control of Network systems*, vol. 2, no. 2, pp. 174–182, 2014.

[15] H. Shen, Y.-J. Pan, U. Ahmad, and B. He, "Pose synchronization of multiple networked manipulators using nonsingular terminal sliding mode control," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2020.

[16] X. Liu, W. Lu, and T. Chen, "Consensus of multi-agent systems with unbounded time-varying delays," *IEEE Transactions on Automatic Control*, vol. 55, no. 10, pp. 2396–2401, 2010.

[17] H. Shen and Y.-J. Pan, "Adaptive robust control of networked multi-manipulators with time-varying delays," in *2019 American Control Conference (ACC)*, pp. 3670–3675, IEEE, 2019.

[18] H. Wei, Q. Lv, N. Duo, G. Wang, and B. Liang, "Consensus algorithms based multi-robot formation control under noise and time delay conditions," *Applied Sciences*, vol. 9, no. 5, p. 1004, 2019.

[19] X. Zhao, X. Zheng, C. Ma, and R. Li, "Distributed consensus of multiple euler–lagrange systems networked by sampled-data information with transmission delays and data packet dropouts," *IEEE Transactions on Automation Science and Engineering*, vol. 14, no. 3, pp. 1440–1450, 2015.

[20] M. Lu and L. Liu, "Leader-following consensus of multiple uncertain euler–lagrange systems subject to communication delays and switching networks," *IEEE Transactions on Automatic Control*, vol. 63, no. 8, pp. 2604–2611, 2017.

[21] U. Munz, A. Papachristodoulou, and F. Allgower, "Delay robustness in non-identical multi-agent systems," *IEEE Transactions on Automatic Control*, vol. 57, no. 6, pp. 1597–1603, 2011.

[22] H. Shen and Y.-J. Pan, "Improving tracking performance of nonlinear uncertain bilateral teleoperation systems with time-varying delays and disturbances," *IEEE/ASME Transactions on Mechatronics*, vol. 25, no. 3, pp. 1171–1181, 2020.

[23] J. R. Klotz, S. Obuz, Z. Kan, and W. E. Dixon, “Synchronization of uncertain euler–lagrange systems with uncertain time-varying communication delays,” *IEEE transactions on cybernetics*, vol. 48, no. 2, pp. 807–817, 2017.

[24] S. Mobayen, “Fast terminal sliding mode tracking of non-holonomic systems with exponential decay rate,” *IET Control Theory & Applications*, vol. 9, no. 8, pp. 1294–1301, 2015.

[25] J. Zhang, M. Lyu, T. Shen, L. Liu, and Y. Bo, “Sliding mode control for a class of nonlinear multi-agent system with time delay and uncertainties,” *IEEE Transactions on Industrial Electronics*, vol. 65, no. 1, pp. 865–875, 2017.

[26] B. Mu, K. Zhang, and Y. Shi, “Integral sliding mode flight controller design for a quadrotor and the application in a heterogeneous multi-agent system,” *IEEE Transactions on Industrial Electronics*, vol. 64, no. 12, pp. 9389–9398, 2017.

[27] D. Nojavanzadeh and M. Badamchizadeh, “Adaptive fractional-order non-singular fast terminal sliding mode control for robot manipulators,” *IET Control Theory & Applications*, vol. 10, no. 13, pp. 1565–1572, 2016.

[28] A. Khanzadeh and M. Pourgholi, “Fixed-time leader–follower consensus tracking of second-order multi-agent systems with bounded input uncertainties using non-singular terminal sliding mode technique,” *IET Control Theory & Applications*, vol. 12, no. 5, pp. 679–686, 2017.

[29] X. Gu, Y. Niu, and B. Chen, “Adaptive non-singular fast terminal sliding mode control for multi-agent systems with unknown non-linear dynamics,” *IET Control Theory & Applications*, vol. 14, no. 16, pp. 2223–2232, 2020.

[30] J. Mei, W. Ren, and G. Ma, “Distributed coordinated tracking with a dynamic leader for multiple euler-lagrange systems,” *IEEE Transactions on Automatic Control*, vol. 56, no. 6, pp. 1415–1421, 2011.

[31] K.-S. Tang, K. F. Man, G. Chen, and S. Kwong, “An optimal fuzzy PID controller,” *IEEE transactions on industrial electronics*, vol. 48, no. 4, pp. 757–765, 2001.

[32] K. Sharma and D. K. Palwalia, “A modified pid control with adaptive fuzzy controller applied to DC motor,” in *2017 International Conference on Information, Communication, Instrumentation and Control (ICICIC)*, pp. 1–6, IEEE, 2017.

[33] M. A. Shamseldin and A. A. EL-Samahy, “Speed control of BLDC motor by using pid control and self-tuning fuzzy PID controller,” in *15th International Workshop on Research and Education in Mechatronics (REM)*, pp. 1–9, IEEE, 2014.

[34] Q. Zhou, P. Shi, S. Xu, and H. Li, “Adaptive output feedback control for nonlinear time-delay systems by fuzzy approximation approach,” *IEEE Transactions on Fuzzy Systems*, vol. 21, no. 2, pp. 301–313, 2012.

[35] C.-C. Hua, Q.-G. Wang, and X.-P. Guan, “Adaptive fuzzy output-feedback controller design for nonlinear time-delay systems with unknown control direction,” *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 39, no. 2, pp. 363–374, 2008.

[36] Z. Song and K. Sun, “Adaptive sliding mode tracking control for uncertain euler-lagrange system,” *IEEE Access*, vol. 7, pp. 56817–56825, 2019.

[37] L. Teng, M. A. Gull, and S. Bai, “Pd-based fuzzy sliding mode control of a wheelchair exoskeleton robot,” *IEEE/ASME Transactions on Mechatronics*, vol. 25, no. 5, pp. 2546–2555, 2020.

[38] B. Chen, X. Liu, K. Liu, and C. Lin, “Fuzzy-approximation-based adaptive control of strict-feedback nonlinear systems with time delays,” *IEEE Transactions on Fuzzy Systems*, vol. 18, no. 5, pp. 883–892, 2010.

[39] K. Erbatur and O. Kaynak, “Use of adaptive fuzzy systems in parameter tuning of sliding-mode controllers,” *IEEE/ASME Transactions on Mechatronics*, vol. 6, no. 4, pp. 474–482, 2001.

[40] 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.

[41] W. Ren and R. W. Beard, “Consensus seeking in multiagent systems under dynamically changing interaction topologies,” *IEEE Transactions on automatic control*, vol. 50, no. 5, pp. 655–661, 2005.

[42] W. Yu, G. Chen, M. Cao, and J. Kurths, “Second-order consensus for multiagent systems with directed topologies and nonlinear dynamics,” *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 40, no. 3, pp. 881–891, 2009.

[43] I. Coonjah, P. C. Catherine, and K. Soyjaudah, “Experimental performance comparison between tcp vs udp tunnel using openvpn,” in *2015 International Conference on Computing, Communication and Security (ICCCS)*, pp. 1–5, IEEE, 2015.

[44] J. Haxhibeqiri, E. A. Jarchlo, I. Moerman, and J. Hoebeke, “Flexible wi-fi communication among mobile robots in indoor industrial environments,” *Mobile Information Systems*, vol. 2018, 2018.

[45] E. W. Weisstein, “Kronecker product.” Accessed on: May 17, 2021. [Online]. Available: `https://mathworld.wolfram.com/KroneckerProduct.html`.

[46] E. W. Weisstein, "Schwarz's inequality." Accessed on: May 17, 2021. [Online]. Available: `https://mathworld.wolfram.com/SchwarzsInequality.html`.

[47] J.-J. E. Slotine, W. Li, *et al.*, *Applied nonlinear control*, vol. 199. Prentice hall Englewood Cliffs, NJ, 1991.

[48] H. K. Khalil, "Nonlinear systems," 2002.

[49] S. P. Bhat and D. S. Bernstein, "Finite-time stability of continuous autonomous systems," *SIAM Journal on Control and Optimization*, vol. 38, no. 3, pp. 751–766, 2000.

[50] J. Heinonen, *Lectures on Lipschitz analysis*. No. 100, University of Jyväskylä, 2005.

[51] J. Liu and X. Wang, *Advanced sliding mode control for mechanical systems*. Springer, 2012.

[52] Y. Feng, X. Yu, and Z. Man, "Non-singular terminal sliding mode control of rigid manipulators," *Automatica*, vol. 38, no. 12, pp. 2159–2167, 2002.

[53] M. H. Zadeh, *Advances in haptics*. BoD–Books on Demand, 2010.

[54] Y.-C. Liu and N. Chopra, "Control of robotic manipulators with input-output delays: An experimental verification," in *Experimental Robotics*, pp. 823–837, Springer, 2014.

[55] Z. Huang, Y.-J. Pan, and R. Bauer, "Leader–follower consensus control of multiple quadcopters under communication delays," *Journal of Dynamic Systems, Measurement, and Control*, vol. 141, no. 10, 2019.

[56] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, 2007.

Appendix A

Author's Publications

C. Cai, Y.-J Pan, S. Liu, and **L. Wan**, "Task space bilateral teleoperation of co-manipulators using power-based TDPC and leader-follower admittance control," in *IECON 2021 The 47th Annual Conference of the IEEE Industrial Electronics Society.* (To be submitted for review).

L. Wan, Y.-J Pan, and H.H. Shen, "Improving synchronization performance of multiple euler-lagrange systems using non-singular terminal sliding mode control with fuzzy logic," in *IEEE/ASME Transactions on Mechatronics*, 2021.

H.H. Shen, Y.-J Pan, and **L. Wan**, "Teleoperated single-master-multiple-slave system for cooperative manipulations in task space," in *Proceedings of the ICPS 2021 IEEE Conference on Industrial Cyberphysical Systems*, 2021.

L. Wan and Y.-J Pan, "Improving performance for multi-agent systems using fuzzy-logic tuning and mixed feedback controller," in *Proceedings of the IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society*, 2020.

L. Wan and Y. -J. Pan, "Analysis of an optical force sensor for haptic applications," in *Progress in Canadian Mechanical Engineering*, 2020.

Appendix B

Journal Paper Copyright Agreement

IEEE COPYRIGHT FORM

To ensure uniformity of treatment among all contributors, other forms may not be substituted for this form, nor may any wording of the form be changed. This form is intended for original material submitted to the IEEE and must accompany any such material in order to be published by the IEEE. Please read the form carefully and keep a copy for your files.

Improving Synchronization Performance of Multiple Euler-Lagrange Systems using Non-Singular Terminal Sliding Mode Control with Fuzzy Logic
Wan, Lucas; Pan, Ya-Jun; Shen, Henghua
Transactions on Mechatronics

COPYRIGHT TRANSFER

The undersigned hereby assigns to The Institute of Electrical and Electronics Engineers, Incorporated (the "IEEE") all rights under copyright that may exist in and to: (a) the Work, including any revised or expanded derivative works submitted to the IEEE by the undersigned based on the Work; and (b) any associated written or multimedia components or other enhancements accompanying the Work.

GENERAL TERMS

1. The undersigned represents that he/she has the power and authority to make and execute this form.
2. The undersigned agrees to indemnify and hold harmless the IEEE from any damage or expense that may arise in the event of a breach of any of the warranties set forth above.
3. The undersigned agrees that publication with IEEE is subject to the policies and procedures of the IEEE PSPB Operations Manual.
4. In the event the above work is not accepted and published by the IEEE or is withdrawn by the author(s) before acceptance by the IEEE, the foregoing copyright transfer shall be null and void. In this case, IEEE will retain a copy of the manuscript for internal administrative/record-keeping purposes.
5. For jointly authored Works, all joint authors should sign, or one of the authors should sign as authorized agent for the others.
6. The author hereby warrants that the Work and Presentation (collectively, the "Materials") are original and that he/she is the author of the Materials. To the extent the Materials incorporate text passages, figures, data or other material from the works of others, the author has obtained any necessary permissions. Where necessary, the author has obtained all third party permissions and consents to grant the license above and has provided copies of such permissions and consents to IEEE

BY TYPING IN YOUR FULL NAME BELOW AND CLICKING THE SUBMIT BUTTON, YOU CERTIFY THAT SUCH ACTION CONSTITUTES YOUR ELECTRONIC SIGNATURE TO THIS FORM IN ACCORDANCE WITH UNITED STATES LAW, WHICH AUTHORIZES ELECTRONIC SIGNATURE BY AUTHENTICATED REQUEST FROM A USER OVER THE INTERNET AS A VALID SUBSTITUTE FOR A WRITTEN SIGNATURE.

Lucas Wan
Signature

10-08-2021
Date (dd-mm-yyyy)

Information for Authors

AUTHOR RESPONSIBILITIES

The IEEE distributes its technical publications throughout the world and wants to ensure that the material submitted to its publications is properly available to the readership of those publications. Authors must ensure that their Work meets the

requirements as stated in section 8.2.1 of the IEEE PSPB Operations Manual, including provisions covering originality, authorship, author responsibilities and author misconduct. More information on IEEE's publishing policies may be found at http://www.ieee.org/publications_standards/publications/rights/authorrightsresponsibilities.html Authors are advised especially of IEEE PSPB Operations Manual section 8.2.1.B12: "It is the responsibility of the authors, not the IEEE, to determine whether disclosure of their material requires the prior consent of other parties and, if so, to obtain it." Authors are also advised of IEEE PSPB Operations Manual section 8.1.1B: "Statements and opinions given in work published by the IEEE are the expression of the authors."

RETAINED RIGHTS/TERMS AND CONDITIONS

- Authors/employers retain all proprietary rights in any process, procedure, or article of manufacture described in the Work.
- Authors/employers may reproduce or authorize others to reproduce the Work, material extracted verbatim from the Work, or derivative works for the author's personal use or for company use, provided that the source and the IEEE copyright notice are indicated, the copies are not used in any way that implies IEEE endorsement of a product or service of any employer, and the copies themselves are not offered for sale.
- Although authors are permitted to re-use all or portions of the Work in other works, this does not include granting third-party requests for reprinting, republishing, or other types of re-use.The IEEE Intellectual Property Rights office must handle all such third-party requests.
- Authors whose work was performed under a grant from a government funding agency are free to fulfill any deposit mandates from that funding agency.

AUTHOR ONLINE USE

- **Personal Servers**. Authors and/or their employers shall have the right to post the accepted version of IEEE-copyrighted articles on their own personal servers or the servers of their institutions or employers without permission from IEEE, provided that the posted version includes a prominently displayed IEEE copyright notice and, when published, a full citation to the original IEEE publication, including a link to the article abstract in IEEE Xplore. Authors shall not post the final, published versions of their papers.
- **Classroom or Internal Training Use.** An author is expressly permitted to post any portion of the accepted version of his/her own IEEE-copyrighted articles on the author's personal web site or the servers of the author's institution or company in connection with the author's teaching, training, or work responsibilities, provided that the appropriate copyright, credit, and reuse notices appear prominently with the posted material. Examples of permitted uses are lecture materials, course packs, e-reserves, conference presentations, or in-house training courses.
- **Electronic Preprints.** Before submitting an article to an IEEE publication, authors frequently post their manuscripts to their own web site, their employer's site, or to another server that invites constructive comment from colleagues. Upon submission of an article to IEEE, an author is required to transfer copyright in the article to IEEE, and the author must update any previously posted version of the article with a prominently displayed IEEE copyright notice. Upon publication of an article by the IEEE, the author must replace any previously posted electronic versions of the article with either (1) the full citation to the IEEE work with a Digital Object Identifier (DOI) or link to the article abstract in IEEE Xplore, or (2) the accepted version only (not the IEEE-published version), including the IEEE copyright notice and full citation, with a link to the final, published article in IEEE Xplore.

Questions about the submission of the form or manuscript must be sent to the publication's editor.
Please direct all questions about IEEE copyright policy to:
IEEE Intellectual Property Rights Office, copyrights@ieee.org, +1-732-562-3966