CONTAINMENT CONTROL BASED ON STATE-FEEDBACK FOR MULTI-AGENTS SYSTEM WITH NON-LINEARITY ELEMENT



UNIVERSITI TEKNIKAL MALAYSIA MELAKA

CONTAINMENT CONTROL BASED ON STATE-FEEDBACK FOR MULTI-AGENTS SYSTEM WITH NON-LINEARITY ELEMENT



This report is submitted in partial fulfilment of the requirements for the degree of Bachelor of Electronic Engineering with Honours

> Faculty of Electronic and Computer Engineering Universiti Teknikal Malaysia Melaka

> > 2022

DECLARATION

I declare that "Containment Control Based on State-Feedback for Multi-Agents System with Non-Linearity Element" is the result of my own work except for quotes as cited in the references.

Farming		JIEM
كل مليسيا ملاك	2	اونيومرسيتي تيڪن
UNIVERSITI TEKNIK Signature	(A	L MALAYZIA MELAKA
Author	:	MAISARAH BINTI AHMAD DAHARI
Date	:	21/06/2022

APPROVAL

I hereby declare that I have read this thesis and in my opinion this thesis is sufficient in terms of scope and quality for the award of Bachelor of Electronic Engineering with Honours.

اونيوم سيتي تبكنيكل مليسيا ملاك
UNIVERSITI TEKNIKAL AALAISA HELAKA
Supervisor Name : DR. AHMAD SADHIQIN BIN MOHD ISIRA

DEDICATION

I dedicate this Final Year Project thesis to my parents, families, supervisor, and friends, who always understand and support me to finish this project and make this project possible. Thank you.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

ABSTRACT

"Multi-agents system" is a term used to describe a group of agents connected to achieve specified control tasks over a communication network. These agents are required to reach an agreement upon certain conditions proposed to reach "containment". This study investigates the containment control problem of the leader-follower configuration in a multi-agents system with a type of non-linearity such as input time delay with respect to a continuous-time and directed spanning forest communication network topology. Control theory and the Laplacian network topology are applied to construct and propose a state feedback containment controller that employs the relative information of each agent. The design controller and containment problem has been achieved in this project where the system is designed with 4 different network topology; Directed, undirected, directed with strongly connected agents and not strongly connected agents. All agents successfully achieved containment through out the system despite the time delay inserted. The stability analysis is done based on Lyapunov stability theory. The goals for this project has been met as all agents able to reach containment and have a stable system. The simulation results of the proposed controllers were shown to prove their theoretical validity. Adding more non-linearities and leaders in one system and applying the current system into a real life project is recommended for future work to improve the system and so that the stability of the system can be proven better.

ABSTRAK

""Sistem berbilang ejen" ialah istilah yang digunakan untuk menggambarkan sekumpulan ejen yang disambungkan untuk mencapai tugas kawalan tertentu melalui rangkaian komunikasi. Ejen-ejen ini dikehendaki mencapai persetujuan atas syarat-syarat tertentu yang dicadangkan untuk mencapai "pembendungan". Kajian ini menyiasat masalah kawalan pembendungan konfigurasi pemimpin-pengikut dalam sistem berbilang ejen dengan jenis bukan lineariti seperti kelewatan masa input berkenaan dengan topologi rangkaian komunikasi hutan rentang masa berterusan dan terarah. Teori kawalan dan topologi rangkaian Laplacian digunakan untuk membina dan mencadangkan pengawal pembendungan maklum balas keadaan yang menggunakan maklumat relatif setiap ejen. Pengawal reka bentuk dan masalah pembendungan telah dicapai dalam projek ini di mana sistem direka bentuk dengan 4 topologi rangkaian yang berbeza; Diarahkan, tidak diarahkan, diarahkan dengan ejen yang berkait rapat dan ejen tidak berkait kuat. Semua ejen berjaya mencapai pembendungan melalui sistem walaupun kelewatan masa dimasukkan. Analisis kestabilan dibuat berdasarkan teori kestabilan Lyapunov. Matlamat untuk projek ini telah dicapai kerana semua ejen dapat mencapai pembendungan dan mempunyai sistem yang stabil. Keputusan simulasi pengawal yang dicadangkan ditunjukkan untuk membuktikan kesahihan teorinya. Menambah lebih banyak bukan lineariti dan peneraju dalam satu sistem dan menggunakan sistem semasa ke dalam projek kehidupan sebenar disyorkan untuk kerja masa depan untuk menambah baik sistem dan supaya kestabilan sistem dapat dibuktikan dengan lebih baik.

ACKNOWLEDGEMENTS

Bismillahirrahmanirrahim. First and foremost, I would like to give thanks and praise to God, the Almighty, for showering me with blessings throughout my research for my Final Year Project and allowing me to complete my thesis on time. I would also like to thanks Dr. Ahmad Sadhiqin bin Mohd Isira, for allowing me to complete my Final Year Project under his supervision and for providing me with invaluable and important assistance during the completion of this project. His dynamism, vision, genuineness, and motivation have impressed me tremendously. It was a distinct honour and privilege to work and study under his supervision. Without his cooperation and expertise, the project would not have reached its end. I am indebted to him for everything. In addition, I would like to thank both of my panels, Dr. Amat Amir and Dr. Mai Maryam, for making my defence a pleasurable experience. My completion of this project could not have been accomplished without the support of my housemates, Lyana, Shafiqah, Izzah and Nisa. Thank you for driving me back and forth to the library and stays until I finished for the day and thank you for always being there whenever I am about to give up and keep pushing me to continue writing until I finally able to submit the completed project. I am also incredibly appreciative of my parents' and siblings' love, prayers, care, and sacrifices for teaching and preparing me for the future. Thank you for being so accepting of my circumstances and the roller-coasters of mood I have been displaying towards you both. Your prayers for me was what sustained me this far. I would also like to thank my fiancé, Ridzwan, for being so patient and for his unwavering support as I conducted research and wrote my project.

TABLE OF CONTENTS

Decl	aration	
Арр	roval	
Ded	ication	
Abs	tract	i
Abs	trak	ii
Ack	nowledgements	iii
Tabl	le of Contents	iv
List	of Tables ALAYSIA	vii
List	of Figures	viii
List	of Abbreviations	Х
List	of Symbols	Х
CHA	PTER 1. INTRODUCTION	
1.1	Brief Overview	1
1.2	Objectives	2
1.3	Problem Statement	3
1.4	Scope of Work	3
1.5	Project Outline	4
CHA	APTER 2 BACKGROUND STUDY	
2.1	Introduction	5
2.2	Mathematical Preliminaries of Consensus Control	5
	2.2.1 Graph Theory	6
	2.2.2 Eigenvalues and eigenvector	7

	2.2.3	Adjacency Matrix	8
	2.2.4	Laplacian Matrix	9
	2.2.5	Kronecker Product	10
	2.2.6	Lyapunov Stability Theorem	10
2.3	3 Controller Design for Single System		11
	2.3.1	State Feedback Controller Design for Single System	11
2.4	Contro	oller Design for Multi-Agents System (MAS)	12
2.5	Time Delay Multi-Agent Systems		13
2.6	Concl	usion	13

CHAPTER 3 METHODOLOGY

3.1	Introduction	14
3.2	Overall Progress	14
	3.2.1 Problem Statement	16
3.3	State Feedback Containment Controller for Multi-Agents System (MAS) without Time Delay	18
3.4	State Feedback Containment Controller for Multi-Agents System (MAS) with Time Delay TEKNIKAL MALAYSIA MELAKA	21
	3.4.1 Stability Analysis State Feedback Containment Controller for Multi-Agents System (MAS)	25
CHA	APTER 4 RESULT AND DISCUSSIONS	
4.1	Introduction	28
4.2	State Feedback Containment Controller for Multi-Agents System (MAS)	28
4.3	.3 State Feedback Containment Controller for Multi-Agents System (MAS) without Time Delay	
	4.3.1 Directed Spanning Tree Connection	30
	4.3.2 Undirected Spanning Tree Connection	33
	4.3.3 Directed Graph with a Strongly Connected Agents	36

	4.3.4	Directed Graph with a Not Strongly Connected Agents	38
4.4	State I with T	Feedback Containment Controller for Multi-Agents System (MAS) Time Delay	41
	4.4.1	Directed Spanning Tree Connection	41
	4.4.2	Undirected Spanning Tree Connection	43
	4.4.3	Directed Graph with a Strongly Connected Agents	46
	4.4.4	Directed Graph with a Not Strongly Connected Agents	48
4.5	Summ	hary of the Chapter	51

CHAPTER 5 CONCLUSION AND FUTURE WORKS

5.1	Conclusion	52
5.2	Future Works	53
REF		54
	اونيومرسيتي تيكنيكل مليسيا ملاك	
	UNIVERSITI TEKNIKAL MALAYSIA MELAKA	

LIST OF TABLES

Table 4.1Connection summary for Multi-agents System (MAS).



51

LIST OF FIGURES

Figure 2.1	Directed Spanning Tree Connection Topology.	7
Figure 2.2	State Feedback Controller Design.	11
Figure 2.3	State Feedback Controller Design for Multi-Agents System.	12
Figure 3.1	Project Flowchart	15
Figure 4.1	Containment Controller State Feedback	29
Figure 4.2	Directed spanning tree communication topology with 3 leaders.	31
Figure 4.3	Simulation of leader-follower configuration states 1 for Directed spanning tree communication topology.	32
Figure 4.4	Simulation of leader-follower configuration states 2 for Directed spanning tree communication topology.	32
Figure 4.5	Undirected spanning tree communication topology with 3 leaders	33
Figure 4.6	Simulation of leader-follower configuration states 1 for Undirected spanning tree communication topology.	35
Figure 4.7	Simulation of leader-follower configuration states 2 for Undirected spanning tree communication topology.	35
Figure 4.8	Directed spanning tree (strongly connected) communication topology with 3 leaders.	36
Figure 4.9	Simulation of leader-follower configuration states 1 for Directed spanning tree (strongly-connected) communication topology.	37
Figure 4.10	Simulation of leader-follower configuration states 2 for Directed spanning tree (strongly-connected) communication topology.	38
Figure 4.11	: Directed spanning tree (not strongly connected) communication topology with 3 leaders.	39
Figure 4.12	Simulation of leader-follower configuration states 1 for Directed spanning tree (not strongly-connected) communication topology.	40

Figure 4.13	Simulation of leader-follower configuration states 2 for Directed spanning tree (not strongly-connected) communication topology.	40
Figure 4.14	Directed spanning tree communication topology with 3 leaders.	42
Figure 4.15	Simulation of leader-follower configuration states 1 for Directed spanning tree communication topology.	42
Figure 4.16	in Simulation of leader-follower configuration states 2 for Directed spanning tree communication topology.	43
Figure 4.17	Undirected spanning tree communication topology with 3 leaders	44
Figure 4.18	Simulation of leader-follower configuration states 1 for Undirected spanning tree communication topology with input delay.	45
Figure 4.19	Simulation of leader-follower configuration states 2 for Undirected spanning tree communication topology with input delay.	45
Figure 4.20	Directed spanning tree (strongly connected) communication topology with 3 leaders.	46
Figure 4.21	: Simulation of leader-follower configuration states 1 for Directed spanning tree (strongly connected) communication topology with input delay.	47
Figure 4.22	Zoomed : Simulation of leader-follower configuration states 2 for Directed spanning tree (strongly connected) communication topology with input delay.	48
Figure 4.23	: Directed spanning tree (not strongly connected) communication topology with 3 leaders.	49
Figure 4.24	: Simulation of leader-follower configuration states 1 for Directed spanning tree (not strongly connected) communication topology with input delay.	50
Figure 4.25	:Simulation of leader-follower configuration states 2 for Directed spanning tree (not strongly connected) communication topology with input delay.	50

LIST OF ABBREVIATIONS

MAS Multi-agents System.



LIST OF SYMBOLS

The Laplacian matrix.

 \mathscr{L}



CHAPTER 1

INTRODUCTION

1.1 Brief Overview

Cooperative Control of Multi-agents System (MAS) has become one of the most fascinating study fields in recent years. Due to its application in evaluating and creating coordinating behaviours among agents in multi-agent frameworks, consensus and containment control in multi-agent systems is becoming progressively popular among researchers. Cooperative control entails working with a group of agents to achieve collective group behaviour via local interaction.

Cooperative control of multi-agent systems has been studied and applied in a variety of settings. Multi-agents are used in a variety of areas, including formation control, flocking, coverage control, project and role assignment, and other cooperative activities. For example, in [1] assessments on flocking control method for UAV groups were explored, as were system uncertainties and unknown disturbance in [2].

According to [3], in a multi-agents system involving the systems and control community, the agents are generally dynamically disconnected from one another, implying the need for cooperation in terms of information sharing between the agents in order to achieve collective behaviour. In this case, each agent requires data from other agents via a direct sensing or communication network, which is often depicted as a graph with nodes. Recently, much progress has been made on both leaderless and leader-following consensus problems (see [4–8] and references therein). Although the leader-following consensus problem for multi-agent systems with a single leader is intriguing, the leaderfollowing problem for multi-agent systems with numerous leaders is sometimes more interesting. The goal of the containment problem is to guide all of the followers into the convex hull spanned by the leaders. Because of its vast applicability in swarm robotics, academics have given a lot of attention to the containment control problem in recent years. [9–12]. In [13], the containment control problem is investigated in both continuous-time and discrete-time domain.

Containment control evolved from consensus control, in which each member is referred to as an agent or subsystem, and the group is referred to as a multi-agent system. Consensus must first be obtained before containment can take place. When the group reaches a consensus, one single outcome is obtained, which is followed by all members or agents in the group. The state feedback containment controller that uses each agent's relative information is developed and proposed using control theory and the Laplacian network structure. According to the design of the controller, the leaders are responsible for creating the convex hull form, which is responsible for containing the followers. For containment to take place, there must be at least one leader who is able to maintain an open line of communication with the followers, whereas in consensus control, there must be no more than one leader at any given time. Following a study of the network structure, the Lyapunov stability theory is referred to in order to derive the stability conditions. The Laplacian matrix must only have a simple zero eigenvalue as one of the containment control's requirements.

1.2 Objectives

- To model the containment control system problem for multi-agents system
- To design state feedback containment controller for multi-agents system
- To analyze the stability of the system with and without the non-linearity elements

• To validate the performance of the controller for a multi-agent system using MATLAB and Simulink

1.3 Problem Statement

Each agent's ability to establish a consensus outcome is one of the issues associated with the multiagent system. The same holds true for containment control. Incorporating the non-linearity factor into the system exacerbates this issue, leading in a non-consensus and non-containment outcome. In recent years, the containment challenge for multi-agent systems has received increasing attention. The issue is met frequently in the real world, such as in distributed computation, flocking, formation flight, and traffic congestion control. A system where the leader is able to contain all the followers by introducing some virtual/actual leaders or containment control system to guide the agents to move about safe places is required to ensure that a group of autonomous agents or robots does not wander into a dangerous location that could result in a non-containment output.

اونيوم سيتي تيڪنيڪل مليسيا ملاك 1.4 Scope of Work TEKNIKAL MALAYSIA MELAKA

To meet the project's objectives, a particular scope must be specified. Modeling a linear multi-agents system using state-space approach is the first step. This model can be achieved by carefully examining the Laplacian matrix generated by the agents' network topology, in which each agent is connected via a directed graph. Before introducing non-linearity features, we conduct a simulation to ensure that the system is operational. Then, aspects of nonlinearity, such as time delay, are introduced into the system, rendering it unstable. A nonlinear containment controller is intended to maintain system stability. The controller's ability to stabilise the system will be validated by experimental simulations.

1.5 Project Outline

This research is divided into five chapters. Chapter 1 briefly describes the background of the study regarding the Nonlinear Containment Control for Multi-Agents System. This section includes the problem statement, project objectives, and scope of work.

Chapter 2 reviews the literature review discussing the mathematical preliminaries of consensus control, and it addresses the mathematical aspects and notations required for consensus control to function correctly. Some preliminaries introduced are graph theory, adjacency matrix, Laplacian matrix, Kronecker product, and the fundamental Lyapunov stability theorem. All these are pointed out in connection to help design the controller.

Chapter 3 is about the methodology. This chapter focuses on the project flow chart of the whole project flow. This chapter also will elaborate more on steps taken to achieve the project results.

Chapter 4 covers all the results that show the step-by-step for the system to reach containment. This chapter will show all the details of the mathematical calculation parts. This chapter also discusses the final results obtained from the previous chapter. The explanation of the results and findings of the project will be discussed in more detail.

Chapter 5 is the final chapter in the thesis, which summarizes all the results and findings from previous chapters. The achievement of this project will be highlighted based on the objectives. Future work will also be included so the improvement can be made for the further research plan.

CHAPTER 2

BACKGROUND STUDY

2.1 Introduction

This chapter discusses the analysis of past researchers' journals in order to improve the project's outcome. There have been a number of publications on this project's relevant topic. A shared variable of interest between agents is required to achieve consensus. In order for containment to be achieved, the system must first reach consensus control. For consensus control to work, it is necessary to understand the mathematical features of consensus. Prior scholars' ideas and publications greatly influenced the concepts of consensus control, the mathematical approach used, and control design. This data was gathered from theses, journals, papers, online books, and other internet sources.

2.2 Mathematical Preliminaries of Consensus Control

A journal on linear multi-agent systems under weighted- balanced and strongly connected digraphs to achieve an optimal point has been discussed in [14]. One of the essential keys in this thesis is to understand how the network topology relates with the controller, and the presence of non-linearities element relates to Lyapunov analysis's stability. The mathematical features of consensus must be explored and understood for consensus control to work thus containment can be reached. Because the mathematical idea of consensus is a well-known subject in the field of computer science, it's easy to see how it's used in consensus control.

Designing a consensus control system entails a number of processes. First of all, a number of agents need to be determined. Then, using Algebraic Graph Theory, the position of each agent is included in the derivation of the adjacency matrix from the graph. The graph is then utilized to generate the Laplacian matrix, which is the core of consensus control. Finally, the system will achieve a consensus output based on numerous definitions and conditions based on this matrix. As a result, this section will focus on Graph Theory and how it pertains to consensus control. However, compared to Computer Science, the amount of material available on this issue is not as large, yet it is sufficient for consensus control.

The first-order (single-integrator) mathematical notion of consensus is studied, followed by the second-order (double integrator) and higher-order (multi-integrator) mathematical concepts of consensus (more than double-integrator). Special algorithms that analyze and use relative state information represent all sorts of consensus. Finally, stability analysis is performed using this data to confirm that the controlled system is stable and that consensus is reached.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2.2.1 Graph Theory

In consensus control, algebraic graph theory is frequently employed to define the topology of agent networking. The graph is based on the topology of information transfer between agents and has a directed or indirect topology. This section explained graph theory's core concept. This also explains the layout of each agent in the group's graph topology to generate the adjacency matrix. The Laplacian matrix will then be produced from the matrix that will be utilised for consensus control.

A directed graph is considered to be strongly connected if there exists a path between every pair of different nodes. A directed graph has a directed spanning tree if there is a node named root (agent 0) such that there is a directed path from this node to every other node, as depicted in the picture 2.1. All work on consensus control of multi-agent systems presented in this study will be based on a directed spanning tree communication topology.



Figure 2.1: Directed Spanning Tree Connection Topology.

Multi-agents must be represented in the graph network in the same manner as other communication networks in order to accomplish the same goal. This topic was examined in depth in [15], focusing on the perspective of graph theory with regard to multi-agent systems, where the attributes of these agents were referred to as action triggers.

2.2.2 Eigenvalues and eigenvector

Eigenvalues and eigenvectors are an important link in the Laplacian matrix for designing consensus control. This section provides a detailed explanation of eigenvalues and eigenvectors [16]. For a square matrix $A = C^{n \times n}$, the eigenvalues of A are defined as the solutions of the equation

$$det(\lambda I - A) \tag{2.1}$$

where I is defined as

$$I = diag(1) \in \mathbb{R}^{n \times n} \tag{2.2}$$

For the set of all the eigenvalues $\{\lambda_i\}_{i=1}^n$, if there is any non-zero vector $vr \in C^n$ satisfying

$$Av_r = \lambda_i v_r \tag{2.3}$$

Then the non-zero vector v_r is called the right eigenvector of A.

Similarly, if a non-zero vector $v_l^T A = \lambda_i v_l^T$ satisfies

$$\boldsymbol{v}_l^T \boldsymbol{A} = \boldsymbol{\lambda}_i \boldsymbol{v}_l^T \tag{2.4}$$

Then the non-zero vector v_l is called the left eigenvector of A. Note that the left eigenvector can also be obtained from the equation $A^T v_l = \lambda_i v_l$

2.2.3 Adjacency Matrix

Corresponding with the communication graph G, adjacency matrix $Q = q_{ij} \in \mathbb{R}^{N \times N}$ is defined as $q_{ii} = 0, q_{ij} = 1$ if $(i, j) \in \varepsilon$ and 0 otherwise. Based on the graph G in Figure 2.1 the adjacency matrix Q can be obtained as

UNIVERSITI TEKNIK

$$Q = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 \\ AL & MALAA \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$
SIA MELAKA

Based on Figure 2.1, the method of producing the adjacency matrix is as follows. The arrow from agent 1 pointing to agent 2 shows the signal flow from agent 1 which give an outcome of 1 in the first column and second row of matrix Q. The signal from agent 5 to agent 1 and vise versa will give outcomes of 1 in column five and rows one and column 1 and row 5 respectively. Agent 1 and agent 2 also receive information from agent 4, which is shown by having outcomes values of 1 in column four and row 1 and row 2. Agent 3 only receives information from agent 2, which results in an outcome of 1 in column 2 and row 3. The value 0 indicates that there is no information transfer.

2.2.4 Laplacian Matrix

The Laplacian matrix is derived from the adjacency matrix Q. it can be defined as

$$\mathcal{L}_{ii} = \sum_{j \neq 1}^{n} q_{ij}$$
$$\mathcal{L}_{ij} = -q_{ij} \text{ for } i \neq j$$
(2.5)

where q_{ij} is defined in Section 2.2.3. From the definition of the Laplacian matrix (2.5) and Figure 2.1, and adjacency matrix Q, the Laplacian matrix \mathcal{L} can be easily obtained as



The Laplacian matrix \mathscr{L} plays an important role in designing the containment controller later in this report. In short, several conditions need to be stated to provide a general understanding of the Laplacian matrix \mathscr{L} . These are the main conditions of the Laplacian matrix \mathscr{L} :

- 1. For a directed graph, \mathcal{L} is not necessarily symmetric.
- 2. since \mathscr{L} has zero row sums, 0 is an eigenvalue of \mathscr{L} with the associated eigenvector $1 \triangleq [1, ..., 1]^T$, and $n \times 1$ column vector of one.
- 3. \mathscr{L} is diagonally-dominant, and has non-negative diagonal entries.

2.2.5 Kronecker Product

The kronecker product of matrix is considered by two matrix $A = [a_{ij}]$ and $B = [b_{ij}]$. The matrix is denoted by $A \otimes B$ which defined by partitioned matrix,

$$A \otimes B = \begin{bmatrix} a_{11}B & \cdots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{m1}B & \cdots & a_{mn}B \end{bmatrix}$$
(2.7)
$$= [a_{ij}B]_{i,\ i=1}^{m,n} \in \mathbb{R}^{mp \times nq}$$

that satisfies the properties mentioned in [17]. Here are some of the useful properties in Kronecker product.



2.2.6 Lyapunov Stability Theorem

The Lyapunov theorem is a fundamental methodology for the stability of the dynamic system, which will be presented in the next chapter. Several publications have dealt with using the Lyapunov stability theorem to solve the consensus problem of multi-agent systems with observer-based protocol [18, 19]. Vepa mentioned in his book [20] Lyapunov's direct method to a linear autonomous system:

$$\dot{x} = f(x) = Ax \tag{2.8}$$

By letting $V(x) = X^T P X$ with $P = P^T > 0$. The derivative of V

$$\dot{V} = \dot{X}^T P X + X^T P \dot{X} = X^T (A^T P + P A)$$
(2.9)

Hence, $\dot{V}(x)$ is negative definite if and only if

$$A^T P + PA + Q = 0 (2.10)$$

By choosing the Lyapunov candidate of $\dot{V} = (\eta^T P \eta)$. A Lyapunov function is ubiquitous when proving the stability analysis in consensus control. In [21, 22], much previous work published in consensus control introduced the same method for their controller to guarantee agents converge.

2.3 Controller Design for Single System

A single system controller will be discussed first and later explained as a multi-agents system as the project will be based on it. When discussing the consensus of multi-agent systems, it is necessary to design a proper controller with an effective method to estimate the system states. By understanding the basics of a single control system, the multi-agent system was introduced with problem formulation in state-space representation, which will be discussed in the next chapter.

2.3.1 State Feedback Controller Design for Single System



Figure 2.2: State Feedback Controller Design.

Consider the state feedback system in Figure 2.2 as follows:

$$\dot{x} = Ax + Bu$$

$$y = Cx \tag{2.11}$$

Where x is the input, A, B and C is are the matrix. K and C are controller gain to be determined later and output. Input in the state feedback system is given by :

$$u = r - Kx \tag{2.12}$$

Hence, by substituting (2.12) into (2.11), the state equation for the state feedback system is represented as:

$$\dot{x} = (A - BK)x + Br$$

$$y = Cx$$
(2.13)

2.4 Controller Design for Multi-Agents System (MAS)

In this section, a controller design for a multi-agents system is introduced to meet the purpose of this thesis. Unlike the previous section, the multi-agents are introduced into the system, which later combined with non-linearity element which is time delay.



Figure 2.3: State Feedback Controller Design for Multi-Agents System.

where i = 1, 2, ..., n represents the subsystems. While x_i , u_i and wx_i are known as state vectors, control input vectors and white noise respectively. By carefully obtaining the Laplacian matrix, *L* from the graph, *G*, these subsystems can be connected From Figure

2.3, the input control is given by:

$$u_i = -K \sum_{j=0}^N l_{ij} x_j \tag{2.14}$$

with *K* as the matrix control gain that will be designed later.

2.5 Time Delay Multi-Agent Systems

Due to the time it takes to transport materials, transmit signals, and so on, delays are unavoidable in physical systems. For a long time, the necessity of addressing delays has been well understood (see [23] and the references therein). For systems with input delays, a wide range of predictor-based methods are useful, including the Smith predictor [24], modified Smith predictor [25], finite spectrum assignment [26], Artstein-Kwon-Pearson reduction method [27], [28], and the shortened predictor feedback approach [29]. For linear systems with any constant input delay, the Artstein-Kwon-Pearson reduction approach is well-known and reasonably uncomplicated. For ordinary differential equations, the stabilisation difficulties are reduced to equivalent problems.

2.6 Conclusion

In conclusion, this chapter discussed the fundamental elements used for this project. First, the properties of graph theories were presented and explained and followed by the concept of eigenvalues and eigenvectors. Next, the flow of matrices analysis made from graph theory into adjacency matrix and Laplacian matrix were presented. Finally, the stability analysis of using the Lyapunov function was explained. In the next chapter, the containment methodology for nonlinear containment control for the multi-agents system will be studied further under directed spanning tree topology.

CHAPTER 3

METHODOLOGY

3.1 Introduction

This chapter is to ensure the containment outcome of the controller design. To deal with the non-linearities, by transforming the Laplacian matrix into Jordan form, stability can be observed with the Lyapunov function. The stability for state feedback controller with Time Delay was also discussed using the respected controller to the system. Next by proving the solution for stability under condition of $P = P^T > 0$ to solve the containment stability. The solution for the stability analysis for state feedback without non-linearity and state feedback controller with time delay is explained in detail in the following sections.

3.2 Overall Progress

Figure 3.1 depicts the overall flowchart for finishing this project, which begins with background studies related to consensus concerns. The multi-agent system is then modelled using the network graph described in the preceding section. The properties associated with the network graph, such as adjacency and Laplacian matrix, will be identified. The conditions for the system were confirmed with mathematical representation after the parameters were correctly determined. After the conditions have been

validated, the consensus analysis will begin. The stability study using the Lyapunov approach is also included in the consensus analysis. After all the consensus analysis has been done and validated, the containment analysis begin and the final results will be verified using simulation.



Figure 3.1: Project Flowchart

3.2.1 Problem Statement

By considering the controller design for N,

$$\dot{x}_i = Ax_i + Bu_i(t-h) \tag{3.1}$$

$$y_i = Cx_i \tag{3.2}$$

where $x_i \in \mathbb{R}^n$ is the subsystem's state vector for i = 0, ..., N, $u_i \in \mathbb{R}^p$ is the *i*th subsystem's input, and $y_i \in \mathbb{R}^q$ is the measured output vector. The appropriate matrices are represented by the matrices $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times p}$, and $C \in \mathbb{R}^{q \times n}$. The h > 0 is the input delay.

A directed graph G defines the subsystem connections. The subsystems are represented by a set of vertices v, while the connections are represented by a set of edges ε . Graph G is related with the adjacency matrix A. If subsystem j is connected to i in G, we get $a_{ij} = 1$. If not, $a_{ij} = 0$.

Next, the Laplacian Matrix (\mathscr{L}), with $\mathscr{L} = \{l_{ij}\}$ can be obtained from the adjacency matrix and is commonly defined by

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

$$l_{ij} = \begin{cases} \sum_{j=1, j \neq 1}^{N} a_{ij} & j = i, \\ -a_{ij}, j \neq i. \end{cases}$$
(3.3)

where the arrangement of the Laplacian matrix $\mathcal L$ is shown as

$$\mathscr{L} = \begin{bmatrix} \mathscr{L}_1 & \mathscr{L}_2 \\ 0_{(N-M) \times M} & 0_{(N-M) \times (N-M)} \end{bmatrix}$$
(3.4)

with $\mathscr{L}_1 \in \mathbb{R}^{M \times M}$ and $\mathscr{L}_2 \in \mathbb{R}^{M \times (N-M)}$.

From the definition of the Laplacian matrix, it is clear that

$$L1 = 0$$

where

$$1 = [1, 1, ..., 1]^T \in \mathbb{R}^N,$$

which implies that the Laplacian matrix has 0 as an eigenvalue associated with the right eigenvector 1.

Few assumptions, definitions and lemmas are needed before continuing to the remaining of this paper.

Assumption 1: Matrices A and B are controllable.

Assumption 2: System dynamic (1) is stable.

Assumption 3: Every single leader subsystem is fixed.

Assumption 4: The communication network G of the multi-agents system contains a directed spanning forest with any of the leaders has a path to the system.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

The Artstein-Kwon-Pearson reduction method is presented. Consider an input-delayed system as 3.1. Let

$$z(t) = x(t) + \int_{t}^{t+h} e^{A(t-\tau)} Bu(\tau-h) d\tau$$
(3.5)

Differentiating z(t) against time yields

$$\dot{z}(t) = Ax(t) + e^{-Ah}Bu(t) + A \int_{t}^{t+h} e^{A(t-\tau)}Bu(\tau-h)d\tau$$
$$= Az(t) + Du(t)$$
(3.6)

Where $D = e^{-Ah}B$. system 3.6 is delay free.

3.3 State Feedback Containment Controller for Multi-Agents System (MAS) without Time Delay

The state-feedback containment controller is proposed as

$$u_i = -K \sum_{j \in \mathscr{F} \bigcup R} l_{ij}(x_i - x_j)$$
(3.7)

The containment control problem is said to be solved if all followers always converge to the stationary convex hull $C_o\{x_j, j \in R\}$ as $t \to \infty$.

For the network dynamic, we have

$$\dot{x} = (I_N \otimes A - \mathscr{L} \otimes BK_C)x \tag{3.8}$$

where \mathscr{L} is defined at (3.3) and (3.4), \otimes is the Kronecker product, $x = [x_f x_l]^T$ where $z_f = [x_1^T, ..., x_M^T]^T$ and $x_l = [x_{M+1}^T, ..., x_N^T]^T$. Hence, we obtain a value for x_f that satisfies the following dynamics:

$$\dot{x}_{f} = (I_{M} \otimes A - \mathcal{L}_{1} \otimes BK_{c})x_{f} - (\mathcal{L}_{2} \otimes BK_{c})x_{l}$$
(3.9)

Let $\xi_i = \sum_{j \in \mathscr{F} \bigcup R} l_{ij}(z_i - z_j), i \in \mathscr{F}$ and we have

$$\xi_f = (\mathscr{L}_1 \otimes I_m) z_f + (\mathscr{L}_2 \otimes I_m) z_l \tag{3.10}$$

Remark 1 Equation (3.10) is derived from

$$\boldsymbol{\xi} = (\mathscr{L} \otimes \boldsymbol{I}_n) \boldsymbol{x} \tag{3.11}$$

or

$$\begin{bmatrix} \xi_f \\ \xi_l \end{bmatrix} = \begin{bmatrix} \mathscr{L}_1 \otimes I_m & \mathscr{L}_2 \otimes I_m \\ 0_{(N-M) \times M} & 0_{(N-M) \times (N-M)} \end{bmatrix} \begin{bmatrix} x_f \\ x_l \end{bmatrix}$$
(3.12)

From (3.8) we can also have

$$\dot{\boldsymbol{\xi}} = (I_N \otimes A - \mathscr{L} \otimes BK)\boldsymbol{\xi} \tag{3.13}$$

Hence, we have

$$\dot{\xi}_f = (I_M \otimes A - \mathscr{L}_1 \otimes BK)\xi_f \tag{3.14}$$

Let's introduce $T \in \mathbb{R}^{N \times N}$ is the non-singular matrices such that



where the Jordan blocks for real eigenvalues $\lambda k > 0$ are represented by $J_k \in \mathbb{R}^{n_k}$ for k = 1, ..., p with the multiplicity n_k is shown as

$$\begin{array}{cccc} \lambda k & 1 \\ \lambda k & 1 \\ & \ddots & \ddots \\ & & \lambda k & 1 \\ & & & \lambda k \end{array}$$

and $J_k \in \mathbb{R}^{2n_k}$ for k = p + 1, ..., q are the Jordan blocks for conjugate eigenvalues $\alpha_k \pm j\beta_k, \alpha_k > 0$ and $\beta_k > 0$, with multiplicity n_k is shown as

$$\begin{bmatrix} \mu(\alpha_k,\beta_k) & I_2 & & \\ & \mu(\alpha_k,\beta_k) & I_2 & & \\ & & \ddots & \ddots & \\ & & & \mu(\alpha_k,\beta_k) & I_2 \\ & & & & \mu(\alpha_k,\beta_k) \end{bmatrix}$$

with I_2 the identity matrix in $\mathbb{R}^{2 \times 2}$ and



Hence, we introduce transformations

$$\dot{\boldsymbol{\eta}} = (T^{-1} \otimes I_n) \dot{\boldsymbol{\xi}} \tag{3.19}$$

where $\boldsymbol{\eta} = [\boldsymbol{\eta}_f \boldsymbol{\eta}_l]^T$. We then obtain

$$\begin{bmatrix} \dot{\eta}_f \\ \dot{\eta}_l \end{bmatrix} = \begin{bmatrix} I_M \otimes A & 0 \\ 0 & I_{(N-M)} \times A \end{bmatrix} \begin{bmatrix} \eta_f \\ \eta_l \end{bmatrix} - \begin{bmatrix} J_f \otimes A & 0 \\ 0 & 0_{(N-M) \times (N-M)} \otimes 0_{M \times M} \end{bmatrix} \times \begin{bmatrix} \eta_f \\ \eta_l \end{bmatrix}$$
(3.20)
From (3.20) we obtain

$$\dot{\eta}_f = (I_M \otimes A - J_f \otimes BK_c)\eta_f \tag{3.21}$$

and

$$\dot{\eta}_l = (I_{N-M} \otimes A)\eta_l \tag{3.22}$$

Hence, in order to further manipulate the special characteristic of \mathscr{L}_1 , (3.38) with respect to η_f is derived as

$$\eta_f = (U^{-1} \otimes I_m)\xi_f \tag{3.23}$$

and (3.21) is used for the design procedure.

tem (MAS) with Time Delay

3.4 State Feedback Containment Controller for Multi-Agents Sys-

The state-feedback containment controller is proposed as

$$(3.24)$$

The containment control problem is said to be solved if all followers always converge to the stationary convex hull $C_o\{x_j, j \in R\}$ as $t \to \infty$.

For the multi-agent system (3.1), we use (3.5) to transform the agent dynamics to

$$\dot{z}_i(t) = A z_i + D u_i(t), \qquad (3.25)$$

by substituting the value of the controller (3.24) into (3.25) we get

$$\dot{z}_i = A z_i - DK \sum_{j \in \mathscr{F} \bigcup R} l_{ij}(z_i - z_j).$$
(3.26)

The closed-loop system is then described by

$$\dot{z} = (I_N \otimes A - \mathscr{L} \otimes DK)z \tag{3.27}$$

where,

$$z = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_N \end{bmatrix}$$

and \mathscr{L} is defined at (3.3) and (3.4), \otimes is the Kronecker product, $x = [z_f z_l]^T$ where $z_f = [z_1^T, ..., z_M^T]^T$ and $z_l = [z_{M+1}^T, ..., z_N^T]^T$. We then introduce a state transformation

$$\xi_i = z_i - \sum_{j=1}^N r_j Z_j$$
 (3.28)



UNIVERSITI TEKNIKAL MALAYSIA MELAKA The dynamics of ξ can then be obtained as

$$\dot{\boldsymbol{\xi}} = (I_N \otimes A - L \otimes DK)\boldsymbol{\xi} \tag{3.29}$$

we then introduce another state transformation

$$\boldsymbol{\eta} = (T^{-1} \otimes I_n)\boldsymbol{\xi} \tag{3.30}$$

then we have

$$\dot{\eta} = (I_N \otimes A - J \otimes DK)\eta \tag{3.31}$$

where

$$\eta = egin{bmatrix} \eta_1 \ \eta_2 \ dots \ \eta_N \end{bmatrix}$$

,

with $\eta_i \in \mathbb{R}^n$ and $\psi : \mathbb{R}^{n \times N} \to \mathbb{R}^n$ for i = 1, 2, ..., N.

Let's introduce $T \in \mathbb{R}^{N \times N}$ is the non-singular matrices such that

$$T^{-1}\mathscr{L}T = J \tag{3.32}$$

With J as a Jordan form block-diagonal matrix with



UNIVERSITITEKNIKAL MALAYSIA MELAKA where the Jordan blocks for real eigenvalues $\lambda k > 0$ are represented by $J_k \in \mathbb{R}^{n_k}$ for k = 1, ..., p with the multiplicity n_k is shown as

$$\begin{array}{cccc} \lambda k & 1 \\ \lambda k & 1 \\ \ddots & \ddots \\ \lambda k & 1 \\ \lambda k & 1 \\ \lambda k \end{array}$$

and $J_k \in \mathbb{R}^{2n_k}$ for k = p+1,...,q are the Jordan blocks for conjugate eigenvalues $\alpha_k \pm$

 $j\beta_k, \alpha_k > 0$ and $\beta_k > 0$, with multiplicity n_k is shown as

$$\mathscr{J}_{k} = \begin{bmatrix} \mathbf{v}(\alpha_{k},\beta_{k}) & I_{2} \\ & \mathbf{v}(\alpha_{k},\beta_{k}) & I_{2} \\ & \ddots & \ddots \\ & & \mathbf{v}(\alpha_{k},\beta_{k}) & I_{2} \\ & & & \mathbf{v}(\alpha_{k},\beta_{k}) \end{bmatrix}$$

with I_2 the identity matrix in $\mathbb{R}^{2 \times 2}$ and

$$\mathbf{v}(\boldsymbol{\alpha}_{k},\boldsymbol{\beta}_{k}) = \begin{bmatrix} \boldsymbol{\alpha}_{i} & \boldsymbol{\beta}_{i} \\ -\boldsymbol{\beta}_{i} & \boldsymbol{\alpha}_{i} \end{bmatrix} \in \mathbb{R}^{2 \times 2}$$
(3.34)

Next, from (3.11), we obtained

$$\begin{bmatrix} \dot{\xi}_{f} \\ \dot{\xi}_{l} \end{bmatrix} = \begin{bmatrix} I_{M} \otimes A & 0 \\ 0 & I_{(N-M)} \times A \end{bmatrix} \begin{bmatrix} \xi_{f} \\ \xi_{l} \end{bmatrix} - \begin{bmatrix} \mathscr{L}_{1} \otimes BK_{c} & \mathscr{L}_{2} \otimes BK_{c} \\ 0_{(N-M) \times M} \otimes 0_{M \times M} & 0_{(N-M) \times (N-M)} \otimes 0_{M \times M} \end{bmatrix} \times \begin{bmatrix} \xi_{f} \\ \xi_{l} \end{bmatrix}$$
(3.35)

Hence, we introduce transformations

$$\dot{\boldsymbol{\eta}} = (T^{-1} \otimes I_n) \dot{\boldsymbol{\xi}} \tag{3.36}$$

where $\boldsymbol{\eta} = [\boldsymbol{\eta}_f \boldsymbol{\eta}_l]^T$. We then obtain

$$\begin{bmatrix} \dot{\eta}_f \\ \dot{\eta}_l \end{bmatrix} = \begin{bmatrix} I_M \otimes A & 0 \\ 0 & I_{(N-M)} \times A \end{bmatrix} \begin{bmatrix} \eta_f \\ \eta_l \end{bmatrix} - \begin{bmatrix} J_f \otimes A & 0 \\ 0 & 0_{(N-M) \times (N-M)} \otimes 0_{M \times M} \end{bmatrix} \times \begin{bmatrix} \eta_f \\ \eta_l \end{bmatrix}$$
(3.37)

From (3.20) we obtain

$$\dot{\eta}_f = (I_M \otimes A - J_f \otimes BK_c)\eta_f \tag{3.38}$$

and

$$\dot{\eta}_l = (I_{N-M} \otimes A)\eta_l \tag{3.39}$$

Hence, in order to further manipulate the special characteristic of \mathcal{L}_1 , (3.38) with respect to η_f is derived as

$$\eta_f = (U^{-1} \otimes I_m) \xi_f \tag{3.40}$$

and (3.21) is used for the design procedure.

3.4.1 Stability Analysis State Feedback Containment Controller for

Multi-Agents System (MAS)

If (A,B) is controllable, we can design the feedback control law for each subsystem as $u_i = -Kx_i$ such that A - BK is Hurwitz. For containment control, one possible control $leiter leiter leiter اونيونر سيتي تيڪنيڪل مليسيا ملاك<math>u_i = -K \sum_{i,j}^N l_{i,j} x_j$ UNIVERSITI TEKNIKAL MJALAYSIA MELAKA law is

(3.41)

where l_{ij} are the elements of the Laplacian matrix L. Note that this control law only depends on the relative information of the system state. Indeed, based on the definition of L, we have

$$u_{i} = -K \sum_{j=1}^{N} \overline{q}_{ij}(x_{i} - x_{j}).$$
(3.42)

where $\overline{q}_i j$ are the elements of the normalised adjacency matrix \overline{q} . With the proposed control law, the closed-loop system is described by

$$\dot{x}_i = A_{xi} - BK \sum_{j=1}^{N} \overline{q}_{ij}(x_i - x_j).$$
 (3.43)

If we use *x* to denote all the subsystem states as

$$x = [x_1^T, \dots, x_N^T]^T$$
(3.44)

we have

$$\dot{x} = (I \otimes A - L \otimes BK)x \tag{3.45}$$

where \otimes denotes Kronecker product. The eigenvalues of the closed-loop system are given by the eigenvalues of the matrices

where
$$\lambda_i$$
 is the ith eigenvalue of *L*.
However, from the definition of *L*, it is easy to see that zero is an eigenvalue of *L* with associated eigenvector $1 = [1, ..., 1]^T$ as we have
UNIVERSITITEKNIKAL MALAYSIA MELAKA
 $L1 = 0.$ (3.47)

Therefore, if the original system matrix A is not Hurwitz, the overall system cannot be asymptotically stabilised by the proposed containment control law (3.41).

The goal of containment control is to ensure that all of the subsystems' state variables are the same. If zero is a simple eigenvalue of L, which is ensured by the requirement that G is highly connected [30], i.e., there is a path connecting every vertex, such a control aim can be realised. This condition, together with the requirement that A_i be Hurwitz, guarantees that the containment control problem can be solved by state feedback.

Indeed, suck a K can always be found if (A, K) is controllable. The consensus control via state feedback is therefore solved if (A, K) is controllable, and G is strongly connected.

Furthermore, it can be shown [31] that x_i converges to

$$r^T \otimes e^{At} x(0) \tag{3.48}$$

where r is the right eigenvector of L corresponding to the simple zero eigenvalue.



CHAPTER 4

RESULT AND DISCUSSIONS

4.1 Introduction

In this chapter, the simulation of the system starts with five agents (1 - 5) using a state feedback controller. The problem statement of the system is shown first without any nonlinear element. Then the simulation of these controllers was shown later in the chapter. It is vital to simulate a state feedback controller without a non-linearity element to show how containment was obtained later with input delay in this project. The containment of each controller is analyzed on how well they reached containment. Containment analysis for multi-agents was done at the beginning of the chapter and followed the containment analysis with input delay presence.

4.2 State Feedback Containment Controller for Multi-Agents System (MAS)

The simulation below was based on five agents. Considering the system is given with state-space representation as follows:

$$\dot{x}_i = Ax_i + Bu_i \tag{4.1}$$

$$y_i = Cx_i \tag{4.2}$$

A state feedback containment control was designed in the Simulink, where it consists of Laplacian subsystems and state feedback subsystems, as shown in Figure 4.1. All the information was updated into the Laplacian matrix subsystem and was feedback into the controller. For each state feedback controller, gain *K* values were also updated controller gain K = B'P and solution for $P = P^T P > 0$ was determined using ARE to maintain the stability, thus making sure the containment was reached. The value for containment later presented in the following section.



4.3 State Feedback Containment Controller for Multi-Agents System (MAS) without Time Delay

This section simulated the containment control for the multi-agents system input delay. The controller used for the simulation was based on the state feedback controller discussed in previous section. The example for the design was based on the statespace representation where matrix A, B and C was assumed. Also, the agents were represented using graph theory and later transformed into Adjacency and Laplacian matrices. Different connection for multi-agents was simulated to observe and analyze whether the containment was able to reach or not.

4.3.1 Directed Spanning Tree Connection

Considering the system above, it has been easily verified that rank (A, C) has a *fullrank* = 2 which means those matrices are stable and controllable. Supposed the system was given with state-space representation with a *N* of agents.



With $P = P^T > 0$ obtained as follows as the solution to the equation ??:

$$P = \begin{bmatrix} 0.4089 & 0.1899\\ 0.1899 & 1.8056 \end{bmatrix}$$
(4.3)



Figure 4.2: Directed spanning tree communication topology with 3 leaders.

Solution for $K = B^T P$ as the controller gain for the system, K was obtained as:

$$K = \begin{bmatrix} 0.1899 & 1.8056 \end{bmatrix}$$
(4.4)

Which showed in (4.7) and (4.8) that all agents converges to zero. Thus, has proven multi-agents system reached consensus with the value of K and P. Same values for control gain K was applied to different configuration of agents and the performances were verified using the Simulink Model as shown in Figure 4.1

UNIVERSITI TEKNIKAL MALAYSIA MELAKA



Figure 4.3: Simulation of leader-follower configuration states 1 for Directed spanning tree communication topology.



Figure 4.4: Simulation of leader-follower configuration states 2 for Directed spanning tree communication topology.

4.3.2 Undirected Spanning Tree Connection

In this section, a multi-agents connection with an undirected spanning tree was simulated using the same system as mentioned in the section. Note that the graph has no specific direction connecting each agents.

$$\dot{x}_i = \begin{bmatrix} -1 & 1 \\ 0 & 0 \end{bmatrix} x_i + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_i$$
$$y_i = \begin{bmatrix} 1 & 0 \end{bmatrix} x_i$$

With $P = P^T > 0$ obtained as follows as the solution to the equation ??:

$$P = \begin{bmatrix} 0.4089 & 0.1899 \\ 0.1899 & 1.8056 \end{bmatrix}$$
(4.5)

Solution for $K = B^T P$ as the controller gain for the system, K was obtained as:

UNIVERSITI TEKNIKAL MALAYSIA MELAKA
$$K = \begin{bmatrix} 0.1899 & 1.8056 \end{bmatrix}$$
(4.6)



Figure 4.5: Undirected spanning tree communication topology with 3 leaders

Considering a different model in this section means that different set of eigenvalues obtained for the Laplacian matrix as 0,0.8299,2.6889,4.0000,4.4812 which were all real distinct integers. Based on Figure 4.17, the Laplacian matrix is shown below

With the values for *K* and *P* obtained from ARE procedure to solve the equation, all agents successfully reached containment using undirected spanning tree. Compared to directed spanning tree, this simulation showed that more agents can be seen as the direction of the agents are not determined as shown in figure 4.6 and 4.7. Therefore, more agents can see their neighbour clearly instead of an agent can only see one agent at a time.



Figure 4.6: Simulation of leader-follower configuration states 1 for Undirected spanning tree communication topology.



Figure 4.7: Simulation of leader-follower configuration states 2 for Undirected spanning tree communication topology.

4.3.3 Directed Graph with a Strongly Connected Agents

In this section, consensus control was based on directed topology but with strong connections where all the agents has at least one connection with the their neighbours.

$$\dot{x}_i = \begin{bmatrix} -1 & 1 \\ 0 & 0 \end{bmatrix} x_i + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_i$$
$$y_i = \begin{bmatrix} 1 & 0 \end{bmatrix} x_i$$

With $P = P^T > 0$ obtained as follows as the solution to the equation ??:



Figure 4.8: Directed spanning tree (strongly connected) communication topology with 3 leaders.

With the eigenvalues has at least one zero and others located in the open right half plane as 3.2720 + 0.0000i, 2.0000 + 0.7862i, 2.0000. Based on Figure 4.20, the Laplacian matrix is shown below.



Figure 4.9: Simulation of leader-follower configuration states 1 for Directed spanning tree (strongly-connected) communication topology.



Figure 4.10: Simulation of leader-follower configuration states 2 for Directed spanning tree (strongly-connected) communication topology.

4.3.4 Directed Graph with a Not Strongly Connected Agents

Directed spanning tree that is not strongly connected was simulated in this section. Using the same system as previous sections, the performance of the multi-agents system without non-linearity element was verified. Based on Figure 4.23, the Laplacian matrix is shown below

$$\dot{x}_i = \begin{bmatrix} -1 & 1 \\ 0 & 0 \end{bmatrix} x_i + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_i$$
$$y_i = \begin{bmatrix} 1 & 0 \end{bmatrix} x_i$$



Figure 4.11: : Directed spanning tree (not strongly connected) communication topology with 3 leaders.



Noticed that in Figure 4.12 and Figure 4.13 the change process of these agents are same as for directed graph with strong connection. Even though the graph was not strongly connected, the agents still converged into containment.



Figure 4.12: Simulation of leader-follower configuration states 1 for Directed spanning tree (not strongly-connected) communication topology.



Figure 4.13: Simulation of leader-follower configuration states 2 for Directed spanning tree (not strongly-connected) communication topology.

4.4 State Feedback Containment Controller for Multi-Agents System (MAS) with Time Delay

4.4.1 Directed Spanning Tree Connection

With the same configuration of agents simulated in containment control for multiagents system without non-linearities, the system was given as:

$$\dot{x}_{i} = \begin{bmatrix} -1 & 1 \\ 0 & 0 \end{bmatrix} x_{i} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_{i}(t - 1.0000e^{-3})$$

$$y_{i} = \begin{bmatrix} 1 & 0 \end{bmatrix} x_{i}$$

Using the same value for solution K and P obtained from ARE procedure for directed spanning tree without non-linearities, the performance of containment control was verified. With the directed spanning tree shown in 4.14, Laplacian matrix obtained was:

Jun all	lo,	4	-		4	ະເ	, n	., w.	نوم	9
4 ⁴ 4 ⁴	1	0	0	0	-1	0	0	0	0	_
UNIVERSITI	$\mathbb{I}_{\mathbb{I}}$	КŅІ	0	0	0	0	S ₀ A	0	0	K A
	0	-1	1	0	0	0	0	0	0	
	-1	0	0	2	0	0	0	0	-1	
$\mathscr{L}=% {\displaystyle\int} \mathcal{L}^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(\mathcal{L})^{2}(L$	0	0	0	0	2	-1	0	-1	0	
	0	0	0	0	0	1	-1	0	0	
	0	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	0	

Noticed that the simulation takes the same process as in multi-agents connection with directed spanning tree without input delay. with the input delay being 0.01*s* the containment output was alike but the containment was not stable as it reach containment. However, all the agents reached consensus under the influenced of input delay. This



Figure 4.14: Directed spanning tree communication topology with 3 leaders.

has proven that with connection directed spanning tree, the agents able to reach containment.

The performances were verified using the Simulink Model as shown in 4.15 and the zoomed-in result is shown in figure 4.16 for clearer view.



Figure 4.15: Simulation of leader-follower configuration states 1 for Directed spanning tree communication topology.



Figure 4.16: in Simulation of leader-follower configuration states 2 for Directed spanning tree communication topology.

4.4.2 Undirected Spanning Tree Connection

5

Considering the multi-agents connection with undirected spanning tree, simulations were done based on the same system where the input and output were as follows,

$$\dot{x}_{i} = \begin{bmatrix} 0 & -1 \\ 0 & 0 \end{bmatrix} x_{i} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_{i}(t - 1.0000e^{-3})$$
$$y_{i} = \begin{bmatrix} 1 & 0 \end{bmatrix} x_{i}$$

Using the same value for solution K and P obtained from ARE procedure for undirected spanning tree without non-linearities, the performance of consensus control was



Figure 4.17: Undirected spanning tree communication topology with 3 leaders

verified. With the directed spanning tree shown in 4.17, Laplacian matrix obtained as:



Applying the same input delay, $h = 1.000e^{-3}$ to the system, the output was given as in Figure 4.18 and Figure 4.19 where all the agents are being contained by the leaders. Noticed that even under the influenced of input delay, all the agents reached containment as the the agents followed the leader's direction and is being contained.



Figure 4.18: Simulation of leader-follower configuration states 1 for Undirected spanning tree communication topology with input delay.



Figure 4.19: Simulation of leader-follower configuration states 2 for Undirected spanning tree communication topology with input delay.

4.4.3 Directed Graph with a Strongly Connected Agents

Different configurations was simulated in this section where multi-agents connected in strong directed spanning tree. The Laplacian matrix was produced based on the network graph in Figure 4.20. The system was considered as follows:

$$\dot{x}_{i} = \begin{bmatrix} -1 & 1 \\ 0 & 0 \end{bmatrix} x_{i} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_{i}(t - 1.0000e^{-3})$$
$$y_{i} = \begin{bmatrix} 1 & 0 \end{bmatrix} x_{i}$$



Figure 4.20: Directed spanning tree (strongly connected) communication topology with 3 leaders.

Using the same value for solution *K* and *P* obtained from ARE procedure for directed spanning tree without non-linearities, the performance of containment control was verified. With input delay h = 0.01 inserted to the input of system, the simulation was obtained as in Figure 4.21 and 4.22 that showed agents reached containment. Even under the influenced of input delay, when the network graph was strongly connected, this gave the effect of stable containment.



Figure 4.21: : Simulation of leader-follower configuration states 1 for Directed spanning tree (strongly connected) communication topology with input delay.



Figure 4.22: Zoomed : Simulation of leader-follower configuration states 2 for Directed spanning tree (strongly connected) communication topology with input delay.

4.4.4 Directed Graph with a Not Strongly Connected Agents

الك

In this last section, the performance of multi-agents connection with not strong directed spanning tree was considered under the influence of input delay. By using the system as follows,

$$\dot{x}_{i} = \begin{bmatrix} -1 & 1 \\ 0 & 0 \end{bmatrix} x_{i} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_{i}(t - 1.0000e^{-3})$$
$$y_{i} = \begin{bmatrix} 1 & 0 \end{bmatrix} x_{i}$$



Figure 4.23: Directed spanning tree (not strongly connected) communication topology with 3 leaders.

Using the same value for solution K and P obtained from ARE procedure for directed spanning tree without non-linearities, the performance of containment control was verified. Based on Figure 4.23, the Laplacian matrix was carefully obtained.



In Figure 4.24 and Figure 4.25, the output showed that all agents reached weak containment under the influenced of input delay.



Figure 4.24: : Simulation of leader-follower configuration states 1 for Directed spanning tree (not strongly connected) communication topology with input delay.



Figure 4.25: :Simulation of leader-follower configuration states 2 for Directed spanning tree (not strongly connected) communication topology with input delay.

4.5 Summary of the Chapter

In this chapter, the containment control for the multi-agents system with and without input delay has been simulated using MATLAB and Simulink. Based on the simulations that have been carried out, all multi-agents under disagreement achieved consensus and is being contained by the leaders assigned, thus containment is also achieved. The results have been simulated using the same set of matrices A, B, and C. Different configurations modelled for the agents which summary of connections is shown in Table 4.1. The containment was successfully achieved using the control gain, K from Hurwitz stability analysis. There were two plots of graphs for each configurations as to represent substates 1 and substates 2. With the input delay used as h = 0.001, showed that the system considered in this paper still able to reach containment.

Type of Connection	Stability	Connection
Directed	strong	One agent see one agent at a time
Undirected	weak	One agent sees other agents and
Allin		also guide its neighbour
Directed (Strongly con-	very strong	All agents receive and transfer data
nected)	Sie	to its neighbour
Directed (not Strongly	weak	Only some agent shares data
connected)	EKNIKAL MAI	AVSIA MELAKA

Table 4.1: Connection summary for Multi-agents System (MAS).

The containment control has been simulated using four graph configurations and all were able to reach an agreement and move towards the same direction. Containment control for the multi-agents system without input delay can reach containment for all configurations, while containment control for the multi-agents system with input delay reached weak containment. However, all agents are able to follow the leader and reached containment within the range of h between 0.001 to 0.3. Note that in this simulation, three specific leader was assigned but the information from neighbours were transferred via Laplacian structure for the system.

CHAPTER 5

CONCLUSION AND FUTURE WORKS

5.1 Conclusion

In conclusion, the objectives of this project have been accomplished. Due to the complexity of developing containment control, multi-agent systems have been one of the most popular study topics, as detailed in Chapter 1. The concept and motivation for the multi-agents system were centred on coordination control, which included convergence and formation control. Based on the work of earlier researchers and publications, the background study of containment control is appropriately provided in Chapter 2. Using the fundamentals of graph theory, the Adjacency and Laplacian matrices were constructed, which aided in determining the eigenvalues. The eigenvalues were very helpful for ensuring that the containment control was stable and that containment could be achieved.

The suggested project is significant to sustainability and ecologically friendly in terms of the efficiency of multi-agent subsystems to conduct a complex task that a single individual is unable to accomplish. Distributing information to several agents can facilitate collective behaviour depending on agent interaction. This technology reduces communication costs and improves the whole system's efficiency, therefore it does not harm the environment. Therefore, it will promote long-term environmental balance.

5.2 Future Works

It is possible to conduct additional investigation into the containment control. With the success of this project's construction of a state feedback containment controller, additional controllers, such as an observer-based controller, can be explored. Other than that, incorporating other non-linearities to the containment control, such as white noise or any other disturbance, can help to advance the field of containment control. A few suggestions can be considered such as:

- 1. Introducing different controller types and inserting other non-linearities to overcome the input delay to produce a better containment among the agents or state.
- 2. Expanding the research for more leaders in one containment control such as 200 leaders in a system for a wider containment control range for multi-agents.
- 3. Apply current system into a real life project with hardware so that the stability of the system can be proven better.

A) ()

Overall, these containment problems can still be improved using appropriate methods and analysis in the future. EKNIKAL MALAYSIA MELAKA

REFERENCES

- [1] P. Yan, C. Bai, H. Zheng, and J. Guo, "Flocking control of uav swarms with deep reinforcement learning approach," 11 2020, pp. 592–599.
- [2] M. Jafari, "A biologically-inspired distributed fault tolerant flocking control for multi-agent system in presence of uncertain dynamics and unknown disturbance," *Engineering Applications of Artificial Intelligence*, vol. 79, 12 2018.
- [3] F. L. Lewis, H. Zhang, K. Hengster-Movric, and A. Das, *Cooperative control of multi-agent systems: optimal and adaptive design approaches*. Springer Science & Business Media, 2013.
- [4] 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.
- [5] W. Ren and R. W. Beard, *Distributed consensus in multi-vehicle cooperative control.* Springer, 2008, vol. 27, no. 2.
- [6] M. Cao, A. S. Morse, and B. D. O. Anderson, "Agreeing asynchronously," *IEEE Transactions on Automatic Control*, vol. 53, no. 8, pp. 1826–1838, 2008.
- [7] Y. Cao, W. Yu, W. Ren, and G. Chen, "An overview of recent progress in the study of distributed multi-agent coordination," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 427–438, 2013.
- [8] M. M. Gulzar, S. T. H. Rizvi, M. Y. Javed, U. Munir, and H. Asif, "Multi-agent cooperative control consensus: A comparative review," *Electronics*, vol. 7, no. 2, 2018. [Online]. Available: https://www.mdpi.com/2079-9292/7/2/22
- [9] M. Ji, G. Ferrari-Trecate, M. Egerstedt, and A. Buffa, "Containment control in mobile networks," *IEEE Transactions on Automatic Control*, vol. 53, no. 8, pp. 1972–1975, 2008.
- [10] Y. Lou and Y. Hong, "Target containment control of multi-agent systems with random switching interconnection topologies," *Automatica*, vol. 48, no. 5, pp. 879–885, 2012. [Online]. Available: https://www.sciencedirect.com/science/ article/pii/S0005109812000830

- [11] J. Li, W. Ren, and S. Xu, "Distributed containment control with multiple dynamic leaders for double-integrator dynamics using only position measurements," *IEEE Transactions on Automatic Control*, vol. 57, no. 6, pp. 1553–1559, 2012.
- [12] G. Wen, Y. Zhao, Z. Duan, W. Yu, and G. Chen, "Containment of higher-order multi-leader multi-agent systems: A dynamic output approach," *IEEE Transactions on Automatic Control*, vol. 61, no. 4, pp. 1135–1140, 2016.
- [13] L. Cheng, Y. Wang, W. Ren, Z.-G. Hou, and M. Tan, "Containment control of multiagent systems with dynamic leaders based on a *piⁿ* -type approach," *IEEE Transactions on Cybernetics*, vol. 46, no. 12, pp. 3004–3017, 2016.
- [14] S. Li, X. Nian, and Z. Deng, "Distributed optimization of general linear multi-agent systems with external disturbance," *Journal of the Franklin Institute*, vol. 358, no. 11, pp. 5951–5970, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0016003221003215
- [15] M. Herrera, M. Pérez-Hernández, A. Kumar Parlikad, and J. Izquierdo, "Multi-agent systems and complex networks: Review and applications in systems engineering," *Processes*, vol. 8, no. 3, 2020. [Online]. Available: https://www.mdpi.com/2227-9717/8/3/312
- [16] P. H. Petkov, T. N. Slavov, and J. K. Kralev, *Design of Embedded Robust Control Systems using MATLAB®/Simulink®*. Control, Robotics and Sensors, 2018, vol. 113.
- [17] A. Poznyak, Advanced Mathematical Tools for Control Engineers: Volume 1: Deterministic Systems. Elsevier, 2010, vol. 1.
- [18] S. Yang, J. Chen, and F. Liu, "Observer-based consensus control of multi-agent systems with input delay," in 2018 IEEE 4th International Conference on Control Science and Systems Engineering (ICCSSE). IEEE, 2018, pp. 533–537.
- [19] A. Isira, N. Hanis, C. Wang, and M. Hairi, "Observer based containment control system for lipschitz nonlinear systems," in 2020 16th IEEE International Colloquium on Signal Processing & Its Applications (CSPA). IEEE, 2020, pp. 296–301.
- [20] R. Vepa, Nonlinear control of robots and unmanned aerial vehicles: an integrated approach. Crc Press, 2016.
- [21] J. Song, K. Li, and C. Hua, "Output feedback consensus control of high-order nonlinear multi-agent systems with full state constraints," in 2019 Chinese Control Conference (CCC). IEEE, 2019, pp. 6212–6217.

- [22] L. Lyubchyk, Y. Dorofieiev, and A. Nikul'chenko, "Consensus decentralized control of multi-agent networked systems using vector lyapunov functions," in 2017 9th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), vol. 1. IEEE, 2017, pp. 60–65.
- [23] J.-P. Richard, "Time-delay systems: an overview of some recent advances and open problems," *automatica*, vol. 39, no. 10, pp. 1667–1694, 2003.
- [24] O. J. Smith, "A controller to overcome dead time," ISA J., vol. 6, pp. 28–33, 1959.
- [25] C. Wang, Z. Zuo, Z. Lin, and Z. Ding, "Consensus control of a class of lipschitz nonlinear systems with input delay," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 62, no. 11, pp. 2730–2738, 2015.
- [26] A. Manitius and A. Olbrot, "Finite spectrum assignment problem for systems with delays," *IEEE transactions on Automatic Control*, vol. 24, no. 4, pp. 541– 552, 1979.
- [27] W. Kwon and A. Pearson, "Feedback stabilization of linear systems with delayed control," *IEEE Transactions on Automatic control*, vol. 25, no. 2, pp. 266–269, 1980.
- [28] Z. Artstein, "Linear systems with delayed controls: A reduction," *IEEE Transactions on Automatic control*, vol. 27, no. 4, pp. 869–879, 1982.
- [29] S. Y. Yoon and Z. Lin, "Truncated predictor feedback control for exponentially unstable linear systems with time-varying input delay," *Systems & Control Letters*, vol. 62, no. 10, pp. 837–844, 2013.
- [30] J. A. Fax and R. M. Murray, "Information flow and cooperative control of vehicle formations," *IEEE transactions on automatic control*, vol. 49, no. 9, pp. 1465– 1476, 2004.
- [31] Z. Li, Z. Duan, G. Chen, and L. Huang, "Consensus of multiagent systems and synchronization of complex networks: A unified viewpoint," *IEEE Transactions* on Circuits and Systems I: Regular Papers, vol. 57, no. 1, pp. 213–224, 2009.