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

KHAIRUN NISA BINTI BAHARUDDIN

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

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

KHAIRUN NISA BINTI BAHARUDDIN

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 Output-Feedback for Multi-agents system with Non-linearity Element" is the result of my own work except for quotes as cited in the references.

Signature : ....

Author : KHAIRUN NISA BINTI BAHARUDDIN

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.

Signature : ....

Supervisor Name : DR. AHMAD SADHIQIN BIN MOHD ISIRA

Date : 22/06/2022

DEDICATION

This thesis is dedicated to my dearly cherished parents and friends, as well as to the greatness of Allah. Not to mention my wonderful supervisor who never stopped teaching me and who never stopped providing me with guidance and motivation to finish this thesis. Also for my brother and sister, who has never left my side, no matter what the situations have been, and has always been there to lend their support. To all of my roommates who inspire me, assist me, give me ideas, and share their knowledge with me. This research is dedicated to all of the special persons in my life who have a profound impact on me and who have shown me love.

ABSTRACT

This research takes into consideration the containment control problem based on the output feedback for the multi-agents system with non-linearity elements for the continuous time period and numerous different forms of spanning tree forest communication topology. When a multi-agent system is contained by a moving leader, each agent's potential to reach a containment outcome is the main problem. In order to keep the system stable, the containment control leader must have control over the follower's blockade. Since the non-linearity elements were added into the model, we built the containment controller for non-linearity elements system to analyze the stability of a system with and without these non-linearity elements. Using MATLAB, verify the system's performance in a simulation.The Hurwitz stabilization must be applied in order to verify that the system has reached a stable state. It is essential to demonstrate, through stabilisation, that the controller system is capable of containing the follower and that the system itself is stable. Stabilizing the system may be achieved using the methodology. Following that, the network structure will be analysed depending on the structure of the network. System stability could be verified using simulation data as an example given. The goals of this research have been met with great success, as the final result is stable and able to reach containment. Some recommendations to improve the system include adding more non-linearity components and leaders, introducing new controllers, apply switching leaders and others.

ABSTRAK

Penyelidikan ini mengambil kira masalah kawalan pembendungan berdasarkan hasil keluaran untuk sistem berbilang agen dengan elemen bukan lineariti untuk tempoh masa berterusan dan pelbagai bentuk topologi komunikasi perhubungan. Apabila sistem berbilang ejen terkandung oleh pemimpin yang bergerak, potensi setiap ejen untuk mencapai hasil pembendungan adalah masalah utama. Untuk memastikan sistem stabil, ketua kawalan pembendungan mesti mempunyai kawalan ke atas lingkungan pengikut. Memandangkan elemen bukan lineariti telah ditambahkan ke dalam sistem, kami membina pengawal pembendungan untuk sistem elemen bukan linear untuk menganalisis kestabilan sistem dengan dan tanpa elemen bukan linear ini. Menggunakan MATLAB, sahkan prestasi sistem dalam simulasi. Penstabilan Hurwitz mesti digunakan untuk mengesahkan bahawa sistem telah mencapai keadaan stabil. Adalah penting untuk menunjukkan, melalui penstabilan, bahawa sistem pengawal mampu mengawal pengikut dan sistem itu sendiri adalah stabil. Menstabilkan sistem boleh dicapai menggunakan metodologi. Selepas itu, struktur perhubungan akan dianalisis bergantung kepada struktur rangkaian. Kestabilan sistem boleh disahkan menggunakan data simulasi sebagai contoh yang diberikan. Matlamat penyelidikan ini telah dicapai dengan kejayaan yang besar, kerana keputusan akhir adalah stabil dan dapat mencapai lingkungan pembendungan. Beberapa pengesyoran untuk menambah baik sistem termasuk menambah lebih banyak komponen bukan linear, memperkenalkan pengawal baharu, memperkenalkan penukaran ketua dan lain-lain.

ACKNOWLEDGEMENTS

To begin, I'd like to convey my sincere thanks to God. since he made it possible for me to complete this thesis and hand it in on time so that I could satisfy the prerequisites for the topic of Bachelor Degree Project (BENU 4972) and the deadline for its submission. In addition, Taking this moment, I'd want to say our gratitude to my advisor, Dr. Ahmad Sadhiqin bin Mohd Isira, for his assistance throughout the duration of the course in the completion of my bachelor degree project. With his guidance, support, and motivation, I was able to accomplish the goals of the project and earn my bachelor's degree. In addition, I am grateful to my manager for his assistance in directing me and spare his important time for me at the vital stages of the project. He also served on the committee and committed a significant amount of his effort and time to the successful completion of this project. Thank you to both my parents and my friends for helping me get this thesis done, and I'd like to extend my gratitude to them as well. That being said, I would want to offer my sincere gratitude to any additional organisation or individual that we did not list but that has given their time and effort to the success of this initiative.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

MAS Multi-Agents System.

LIST OF SYMBOLS

L	Laplacian Matrix.

CHAPTER 1

INTRODUCTION

1.1 Introduction

The field of control system research has paid a large amount of attention in recent decades to a sort of cooperative control known as consensus control. This type of control has garnered a significant amount of attention in recent years. It is a control action for interconnected multi-agents systems that rely on relevant data of each agent in the network neighbourhood in terms of reaching agreement [1]. This idea comes from the field of computer science. It is based on a notion. The term "network consensus" refers to a situation in which the states or outputs of all agents that are exposed to a particular communication network topology converging to specified quantities of interest [2]. When all of the subsystems are controlled to accomplish the same control objective, a control system has what is also known as consensus control.It was the primary goal of consensus for multi-agent systems to bring them all together in a single state. As a group, they came up with a decision value. The decision making value was not determined by centralised systems, but rather by each agent using its own and neighbouring information [3].Consensus output is another name for the result of such a control system.

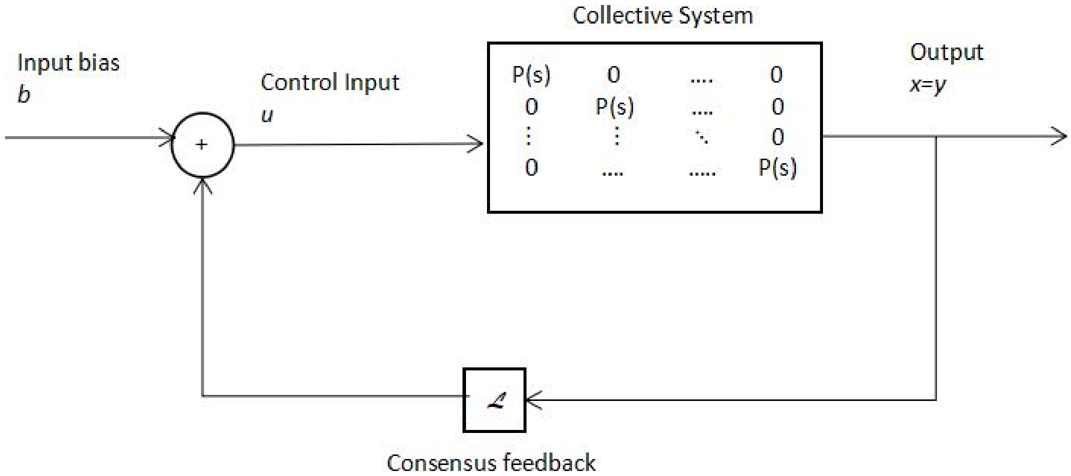

Figure 1.1: The concept of consensus control.

In [4] mentioned that the problem is a result of communication delays between leaders and their subordinates. There are three types of communication delays to recognise: uniform, non - uniform, and time-varying.A great deal of attention has been devoted to the concept of consensus control as a kind of cooperative control over the past decade. Its design typically concentrates on the communication structure, which is indicated by a unique structure known as the Laplacian structure. This is done so that each dynamical subsystem in networks with a swapping or fixed connection can accomplish the same or similar objectives or responsibilities.

When it comes to the design and analysis of consensus control systems, the connections between the subsystems are quite significant. Concepts from graph theory and control theory can be utilised to perform an analysis of the system's stability. A structure that is exclusive to the information flow between subsystems has been modelled after the tree structure of a communication network. When adopting multi-agent consensus control, all of the focus is placed on a single subsystem that has dynamics that are analogous to those of the other subsystems. This causes all of the subsystems to carry out the same action.

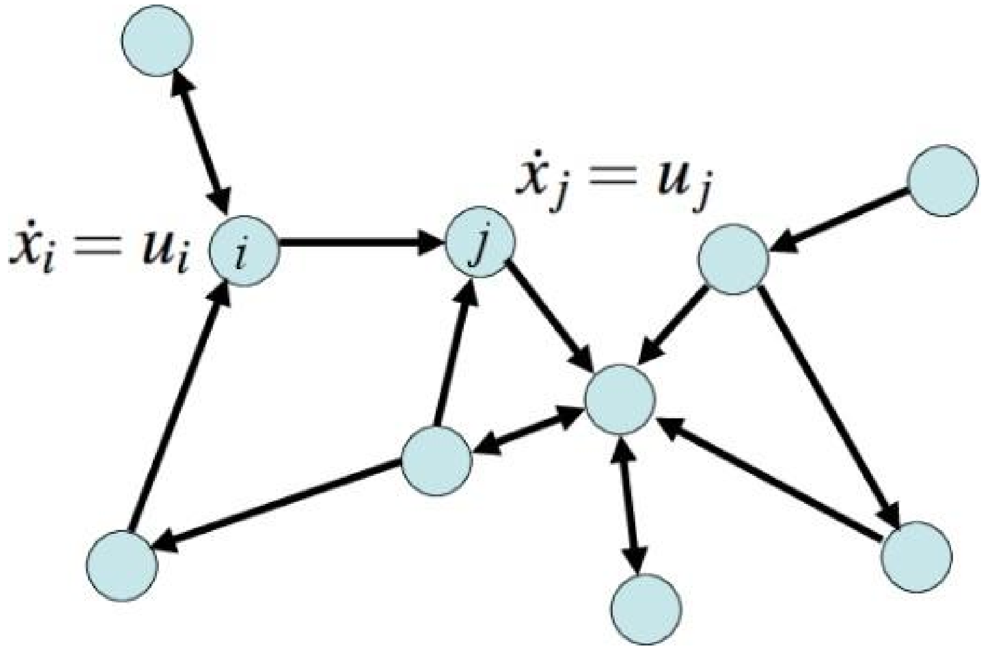

Figure 1.2: The network of Multi-Agents System (MAS).

There are a wide variety of additional applications to make use of consensus methods, including formation control, synchronisation, and others that have been developed more recently. Skills in cooperative control are utilized in various applications including such job, formation control, Cooperative search, air-craft traffic control, swarming, flocking,and another communication system [5]. Other examples of these types of applications include: Flocking and the regulation of flocking are two more instances that come to mind [6]. These applications can be categorised into three distinct kinds of data transfer among both agents: 1^{st}order systems (single-integrator), 2^{nd} order systems (double-integrator), and higher-order systems, all of which have received a significant amount of attention in recent years. Higher-order systems are the most recent type of information transfer between agents to receive a great deal of focus.

Figure 1.3: The flock of birds.

The containment control can be seen as an extension of consensus control. Unlike consensus control,in which multi-agent systems typically operate with a single leader, involves several leaders who work in a forest connection topology. The result of such a control technique is the containment of the followers by the leaders. These controllers make use of the relative state information as well as the relative output information that is provided by each agent or subsystem.

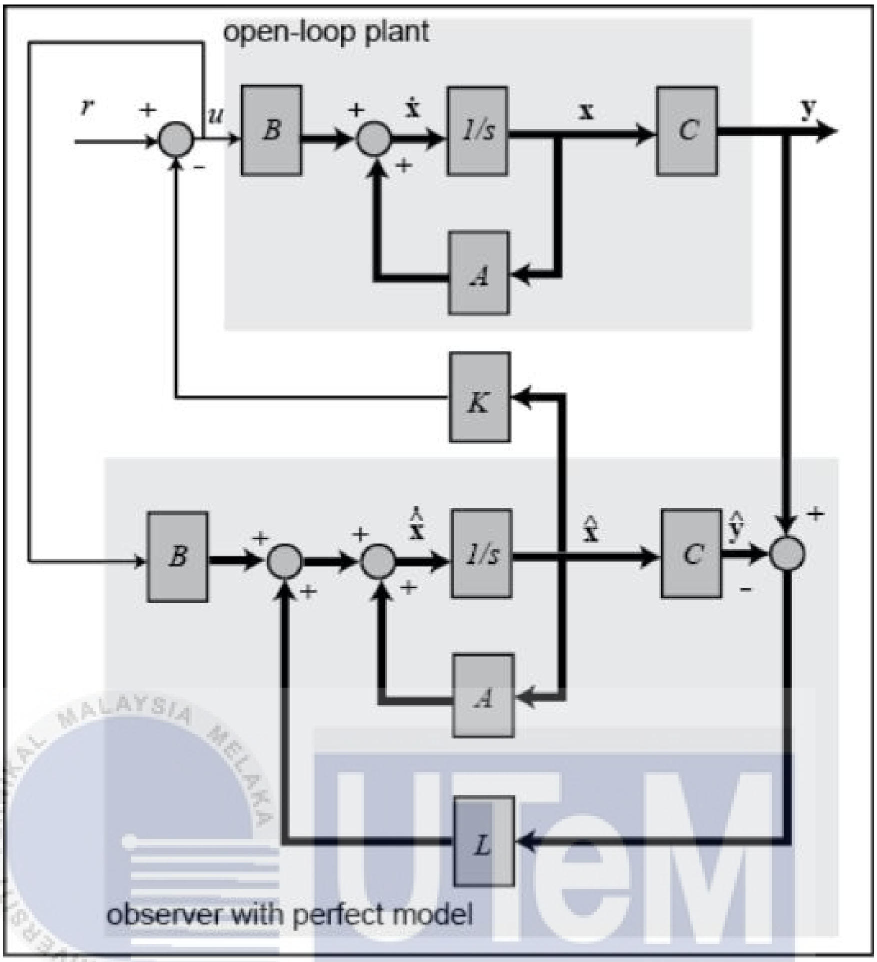

Figure 1.4: Observer model.

While observing only the output , an observer can be constructed to estimate all of the state variables when this isn't possible. There are three new predicted state variables in the magnetic ball scenario.The observer is essentially a duplicate of the plant, which has the same input and nearly a same differential equation [7].The observer is the existence scenario has defined a set of conditions that must be met in order for it to exist, but these conditions are incredibly difficult to meet in practise. To acquire the real transformation function, one must first evaluate a set of simultaneous dynamical systems. Even if these prerequisites are met, building the observer will still be difficult. In order to stabilise a completely linearizable nonlinear system, an observer-based controller is built to make extra assumptions about the plant [8].

1.1.1 Problem Statement

The primary issue arises in the containment control of multi-agent systems when these systems are contained by many leaders who move, hence hindering the ability of individual agents to arrive at a containment result. In order to keep the system from becoming unstable, the leader of the containment control must participate in the blockade of the follower.

1.1.2 Objective

i. To model the multi-agents system with non linearity elements.

ii. To design the containment controller for a multi-agents system with non linearity elements.

iii. To analyze the stability of the system with controller and without controller.

iv. To verify the outcome of the project with simulation in Matlab/Simulink.

1.1.3 Scope of Project

Firstly, the multi-agents system need to be modeled with the state-space methodology and be simulated. Then, the observer and controller been designed Analyzed to check the stability of the system with & without the controller based on the stability condition. The stability of the system will analyzed by using the Lyapunov stability. From this stability analysis, the value of controller gain and obserber gain (k and L) will be determined. To validate the system using value of k and L, the simulation is run using the MATLAB.

1.2 Thesis Organization

There are a total of five sections to this dissertation. There is an overview of consensus control, the purpose of the research and the scope of the project in the first chapter. This introduction chapter is followed by a background study on the single and multi-agent systems. Consensus control is also covered, as are the mathematical elements and notations that are necessary for it to function. The adjacency matrix, the Laplacian matrix, and the Kronecker product are all parts of graph theory.

In addition, the Lyapunov stability theorem is discussed in relation to the creation of a controller and observer for a single linear system in this chapter. There are three types of first-order (single integrator), second-order (double integrator), and higher-order consensus controls described mathematically in the chapter, with the most current papers for each category. The containment observer for a time delay system and publications on the stability analysis of state-feedback consensus containment control. The ability to monitor and respond to output. The Lyapunov stability analysis is also used in this system's analysis.The project's approach, presented in Chapter 3, explains how the project's work is organised and progresses.Observer-based controllers with and without a time delay are also discussed in this chapter in comparison to the results of a consensus observer. Then publications draw conclusions based on the comparison of the results of the consensus observer and the observer-based controller.

The next finding in this thesis is detailed in Chapter 4. Consider the fact that this thesis's main goal is to find a way to keep a multi-agent system with nonlinear aspects contained through output feedback. As a result, this chapter proposes two confinement controllers: one based on state input and the other on observers. The observer-based containment controller uses the consensus observer, which is based on Luenberger's observer.

An in-depth explanation of how the Laplacian matrix (L) structure differs from system output when choosing between consensus and confinement control strategies is provided. The stability of the system is examined again using the Hurwitz stability analysis. In order to demonstrate that the controller is working properly, the next section includes simulation examples with thorough explanations. To ensure that the containment is achieved as illustrated in the simulation section, the controller gain K is determined from Hurwitz stability.

For Multi-Agents System (MAS) with non-linearity features, the confinement control based on output feedback is shown in Chapter 5. All of the simulations have been run and thoroughly examined. There is a concluding paragraph that summarises all of the previous chapters' findings and results. Additional information is provided in this chapter on how to use output feedback to regulate multi-agent systems in the future.

CHAPTER 2

BACKGROUNG STUDY

2.1 Introduction

According to previous articles, the primary goal is to achieve consensus without any leader-follower structure. The leader-follower structure with a single leader who can ensure all agents converge in a blockage of the leader to achieve a specific point or value in a specific place or region has also been found to be an effective strategy.The notion of containment control for MAS, which calls for the states that are considered to be followers to converge toward the convex hull produced by the states that are considered to be leaders [9]. In addition, when the same leader-follower configuration increases, there are multiple leaders. They are able to handle their following among leader and follower by supplying a particular amount of bound. Furthermore, this control action is referred to as a containment control system or a consensus extender. A fixed leader or dynamic, however, created a convex hull by converging an agent to a specific location. Using the multiple leader-follower network topology, this convex hull may be generated. Previous studies have looked at numerous ways to solve this containment problem mathematically.

There are characteristics that enables all followers to converge in a hull formed by the directed graph structure and stationary leaders, which may be applied to any finite-dimensional system with fixed values. In [10], Within the context of cooperative control and finite-time Lyapunov function theory, an observer-based adaptive fuzzy finite-time output-feedback containment control system is developed with the help of an adaptive backstepping control design technique and an integral compensator approach.

2.2 Consensus Control with nonlinearity

This system included the matrix as an integral part of the entirety of the value that it produced in order to function as a consensus control system [11]. The system that is being investigated is made up of a connection between seven different subsystems and a leader, and every one of these subsystems is described by a 2^{nd} order state space model as follows [12]:

$$\dot{x}_i = Ax_i + Bu_i \tag{2.1}$$

The Consensus condition :

1. Laplacian matrix condition must have one simple 0.
2. The system must stable (use hurwitz stability)

In order to discover the location of the containment, one must first discover the location of the consensus. Mathematically speaking, each agent in the multi-agents network system is delegated to either remain in or converge to a space that is referred to as a convex hull. This space is founded by the dynamic or static leaders in the system, and it is the space that the agents are expected to remain in or move toward.

This space may have been derived from the system's multiple leader-follower link architecture in some way. The absence of zero-valued elements in the left eigenvector with respect to the eigenvalue of the Laplacian matrix is another another circumstance that allows for the possibility of containment.

The connection between subsystem shown

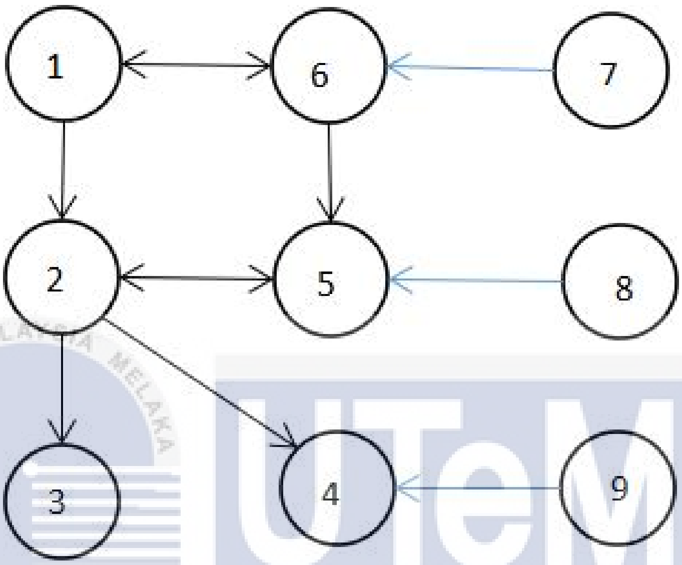

Figure 2.1: The connection between subsystem.

leaders (7–9) and followers (1–6), with the dotted lines denoting connections between at least one leader and one follower at any one moment.

The nonlinearity that was utilised in the control design in order to cope with the bounded control input from the leader. (extensive noise; temporal delay; parallelism). The system for connecting to the network, with each individual subsystem exhibiting the same dynamic and carrying out the same function [13].

Each individual subsystem makes use of the information that is sent to it over the network.The connectivity topology of the multiagent system is described using a directed graph [14] [15].It should be noted that the construction of the static controller necessitates knowledge of the Laplacian matrix's eigenvalues as well as the upper bounds of the leaders' controlling inputs [16].

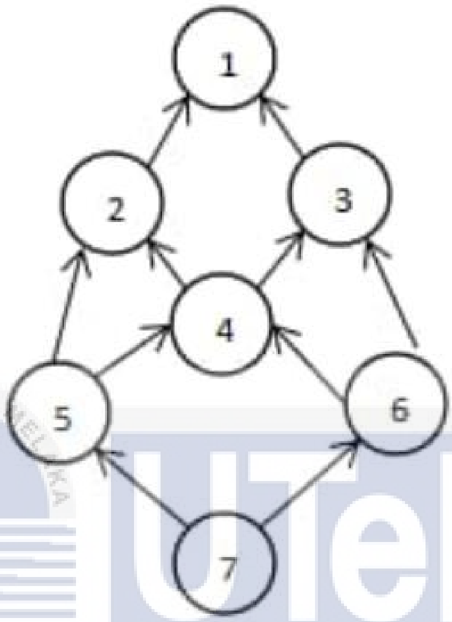

Figure 2.2: The connection between subsystem.

The eigenvalue of L are given as 2, 2, 2, 2, 1, 1 which can be differentiated. The Jordan Matrix (J) matching the criteria of L can be obtained as

$$J = \begin{matrix} 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{matrix}$$

2.3 Multi-agents consensus control with non linearity

$$\dot{x}_i = \begin{bmatrix} -1 & -1 \\ 1 & 0 \end{bmatrix} x_i + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_i + \gamma \begin{bmatrix} sin(Cx_i) \\ 0 \end{bmatrix} \tag{2.2}$$

and

$$C = \begin{bmatrix} 1 & 0 \end{bmatrix} \tag{2.3}$$

The connection between subsystem shown in figure :

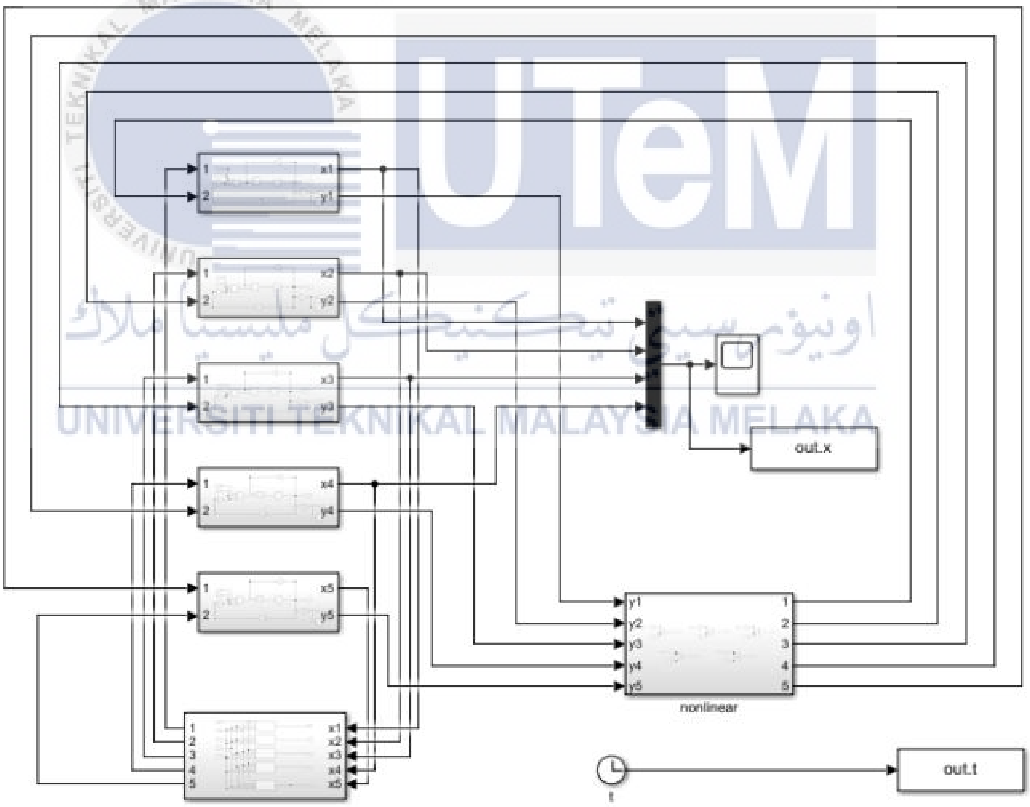

Figure 2.3: Multi-agent with nonlinearity.

2.4 Graph Theory

The topology of information transit between agents serves as the foundation for the graph, which combines directed and undirected topologies to generate its overall structure [17]. In order to adequately characterise the topology of the networking between agents, algebraic graph theory is frequently employed.In order to tackle the problem, Laplacian-based consensus matrices are constructed using a time-invariant undirected graph, which simulates the interactions between nodes in the network. [18]. This is the rationale behind the graph topology arrangement of each participant in the group that forms an adjacency matrix [19]. After that, the Laplacian matrix that will be utilised in consensus control will be derived from the matrix.

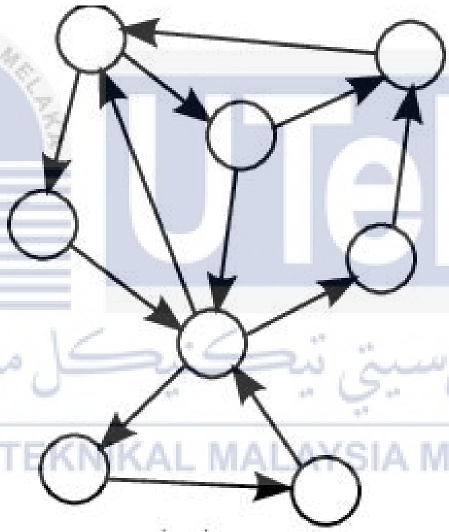

Figure 2.4: Directed Graph.

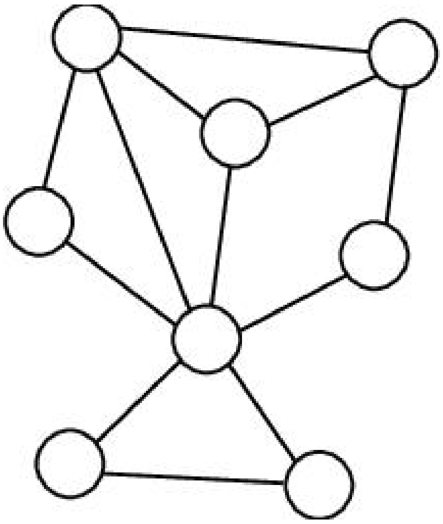

Figure 2.5: Undirected Graph.

2.4.1 Eigenvalues and Eigenvectors

The ideas of eigenvalues and eigenvectors are broken down in comprehensive detail in the following paragraphs. When developing a consensus control scheme, the Eigenvalues and Eigenvectors of the Laplacian matrix can prove to be very helpful links [20]. In the case of the square matrix, the solutions to the equation are specified to be the eigenvalues of the matrix A.

$$\det(\lambda I - A) \tag{2.4}$$

Where I is defined as

$$I = diag(1) \in \mathrm{R}^{nxn} \tag{2.5}$$

For the set of eigenvalues λ_i for $i = 1 \ldots n$, if there is any non-zero vector $v_r \in C^n$ satisfying

$$Av_r = \lambda_i, v_r \tag{2.6}$$

Them 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

$$v_l^T A = \lambda_i v_l^T \tag{2.7}$$

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

2.4.2 Gershgorin Circle Theorem

In mathematics, the Gershgorin circle theorem is usually used to bound the eigen-value of a square matrix i.e the Laplacian matrix [21]. A square matrix $A \in C^{nxn}$ is defined in which the elements of matrix A are a_{xy} for $x, y = 1, 2, , n$. Let

$$R_x = \sum_{x=1, x \neq y}^{n} a_{xy} \tag{2.8}$$

which is the sum of the x^{th} rows of the matrix A except the elementa_{xy}.The Gershgorin circle theorem pointed out that the eigenvalues of matrix A will located in the closed discs, which are centred at a_{xy} with radius R_x. are also called Gershgorin discs.

2.4.3 Adjacency Matrix Q

The adjacency matrix Q is derived from the graph G mentioned in the previous section. For example, suppose that there are N nodes in a graph: for this thesis, the adjacency matrix $Q = (q_{ij} \in \mathrm{R}^{NXN})$ is defined by $q_{ij} = 0$, $q_{ij} = 1$ if $(i, j) \in \varepsilon$, and 0 otherwise [22].

2.4.4 Laplacian Matrix

The L , with $L \in \mathrm{R}^{NxN}$ is derived from the adjacency matrix Q. It can be defined as

$$L_{ii} = \sum_{j \; i}^{N} q_{ij} \tag{2.9}$$

$$L_{ij} = -q_{ij} \text{ for } i \neq j \tag{2.10}$$

where q_{ij} is defined. From the definition of the Laplacian matrix and adjacency matrix Q, the Laplacian matrix L can be easily obtained as

$$L = \begin{matrix} 2 & 0 & 0 & -1 & -1 \\ -1 & 2 & 0 & -1 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 \\ -1 & 0 & 0 & 0 & 1 \end{matrix} \tag{2.11}$$

The Laplacian matrix L is from the directed topology, based on Consequently, Laplacian matrix L can be obtained as

$$L = \begin{matrix} 3 & -1 & 0 & -1 & -1 \\ -1 & 3 & -1 & -1 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ -1 & -1 & 0 & 3 & -1 \\ -1 & 0 & 0 & -1 & 2 \end{matrix} \qquad (2.12)$$

It is plain to see that the topology of undirected connections is essentially a specialised form of the topology of directed connections [23]. The Laplacian matrix that was created is approximately normal, and each data transmission in the line G truly possesses the same characteristics as the input signals of agent 1 through agent 4. In later sections of this study, the The Laplacian matrix L will play a significant part in the process of developing the consensus controller and observer. In a nutshell, it is necessary to describe a number of requirements in order to offer a comprehensive comprehension of the Laplacian matrix L . The following is a list of the primary criteria of the L , each of which plays a vital part in the process of developing the consensus controller:

1. For an undirected graph, L is symmetric.
2. For a directed graph, L iscertainly often not symmetrical.
3. Directed and undirected situations are both examples of this. Since L has zero row sums, 0 is an eigenvalue of L along with the related eigenvector 1 , $[1, \ldots, 1]^T$, and $n1$ column vector of ones.
4. L is diagonally-dominant, and has non-negative diagonal entries.
5. Gersgorin disc theorem states that, for an undirected graph, all of the non-zero eigenvalues of L are positive (L is positive semidefinite), however, each of L eigenvalues that are not zero have positive real portions in the case of a directed

graph. Since all of L eigenvalues are non-zero, all of L real components are negative.

6. For an undirected graph, 0 is a simple eigenvalue of L if - and only if - the undirected graph is connected.

7. It is possible to have a directed graph that has strong connections, but the connection doesn't remain true for an undirected graph, it is possible to have a strongly linked graph, but the connection doesn't hold true.let $\lambda_i(L)$ be the i_{th} smallest eigenvalue of L with $\lambda_1(L) \leqslant \lambda_2(L) \leqslant \quad \leqslant \lambda_n(L)$ so that $\lambda_1(L) = 0$

8. For an undirected graph, $\lambda_2(L)$ has to do with the algebraic connection, which is only positive when an undirected graph is connected. Algebraic connectedness can be used to quantify the convergence rate of consensus algorithms.

2.4.5 Kronecker Product

The Kronecker product of matrices $A \in \mathrm{R}^{mxn}, B \in \mathrm{R}^{pxq}, C \in \mathrm{R}^{rxs}$ and $D \in \mathrm{R}^{txu}$ is defined as

$$A \otimes B = \begin{matrix} a_{11}B & \dots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{m1}B & \dots & a_{mn}B \end{matrix} \tag{2.13}$$

that satisfies

$$(A \otimes B)(C \otimes D) = (AC) \otimes (BD)(A \otimes B)^T = A^T \otimes B^T(A \otimes B) + (A \otimes C) = A \otimes (B + C)$$

2.4.6 Lyapunov stability theorem, controller and observer designs

The Lyapunov stability theorem was utilised in order to ascertain whether or not the dynamics system was stable. Equally autonomous and non-autonomous systems can use it, with the autonomous system being defined by the Lyapunov stability theorem. This theorem is utilised to establish whether or not a dynamical system is stable [24] [25]. It can be used for autonomous systems as well as non-autonomous systems, with autonomous systems being defined as follows: :

$$\dot{x} = f(x) \tag{2.14}$$

For the observer design used when we are unable to measure all of the state variables x, we turn to the observer to provide an estimate of those values. while solely focusing on measuring the output $y = Cx$. In addition to this, it is a replica of the plant with an the difference between the actual and estimated output, plus another factor, is called the $\hat{y} = C\hat{x}$. This will contribute to the correction of the estimate $\hat{x}$ and bring it closer to the real state's values. (provided that the inaccuracy in the measurement is not too great)

$$\dot{\hat{x}} = A\hat{x} + B_u + f(y - \hat{y}) \tag{2.15}$$

$$\hat{y} = C\hat{x} \tag{2.16}$$

The observer's error dynamics are determined by $A - LC$

$$\dot{e} = \dot{x} - \dot{\hat{x}} = (A - LC)e \tag{2.17}$$

To get started, you are going to have to choose the observer gain (L). We need to locate the poles at least 5 times further towards the left than that of the dominant poles of the system and we want the dynamics of the observer to be considerably quicker than the dynamics of the system itself. Because of this, we will be able to attain the greater dynamics that we have been working toward.

If we want to make the most of the space that we have, we will need to set comprising three observer pole in different locations throughout the room.Because controllability and observability are two sides of the same coin, we can utilise the same method that was used to obtain the control matrix by exchanging the matrix B for the matrix C and then taking the transposes of both matrices. This will give us the control matrix. For feedback, we make use of the predicted state, denoted by , $u = -K\hat{x}$. Adding the following lines to your m-file will allow you to see the behaviour of the system in response to a non-zero initial condition when there is no reference input. In this section, we will make the assumption that the observer starts off with an initial estimate of zero. Because of this, the initial prediction error will be close to the actual state vector, which will be denoted by the notation $e = x$.

2.5 The mathematical of Observer based controller

In this section, explain how Observer-based controller works mathematically. Based on what we gained in the last section about the Laplacian equation and the stability theorem, we can proceed on and propose that State-feedback Consensus Containment Control.

2.5.1 State-feedback Consensus Containment Control

The state-feedback containment consensus controller is proposed as

$$u_i = -K \sum_{j \in F \cup R} l_{ij} (x_i - x_j) \tag{2.18}$$

where $K \in R^{pxn}$is a gain control matrix that contains a constant and will be generated at a later time. It would be possible to say that the containment consensus control problem has been solved if all of the followers constantly converged to the stationary convex hull.. $C_o \{x_j, j \in R$ as $t \to \infty$

For the network dynamics, we have

$$\dot{x} = (I_N \otimes A - L \otimes BK)x \tag{2.19}$$

where L is defined, $\otimes$ is the Kronecker product, $x = [x_f \; x_l]^T$ where $x_f = [x^T{}_1, \ldots\ldots\ldots, x_M^T]^T$ and $x_l = [x_{M+1}^T \ldots, x_N^T]^T$. Hence, we are able to derive a value for x_f that conforms to the following dynamic requirements :

$$\dot{x}_f = (I_M \otimes A - L_1 \otimes BK)x_f - L_2 \otimes BK)x_1 \tag{2.20}$$

Let us reintroduce nonsingular matrices $T \in \mathrm{R}^{NXN}$ and $T^{-1} \in \mathrm{R}^{NXN}$ such thatwith

J being a block-diagonal matrix of real Jordan form

$$J = \begin{bmatrix} J_1 & & & & & & \\ & J_2 & & & & & \\ & & \ddots & & & & \\ & & & J_p & & & \\ & & & & J_{p+1} & & \\ & & & & & \ddots & \\ & & & & & & J_q \end{bmatrix} \tag{2.21}$$

where $J_k \in \mathbb{R}^{n_k}$ for $k = 1, \ldots..p$ are the Jordan blocks for real eigenvalues $\lambda_k > 0$, with the multiplicity n_k in the form

$$J_k = \begin{bmatrix} \lambda_k & 1 & & & \\ & \lambda_k & 1 & & \\ & & \ddots & \ddots & \\ & & & \lambda_k & 1 \\ & & & & \lambda_k \end{bmatrix} \tag{2.22}$$

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

$$J_k = \begin{bmatrix} \mu(\alpha_k, \beta_k) & I_2 & & & \\ & \mu(\alpha_k, \beta_k) & I_2 & & \\ & & \ddots & \ddots & \\ & & & \mu(\alpha_k, \beta_k) & I_2 \\ & & & & \mu(\alpha_k, \beta_k) \end{bmatrix} \tag{2.23}$$

with I_2 the identity matrix in $\mathbb{R}^{2x2}$ and

$$\mu(\alpha_k, \beta_k) = \begin{bmatrix} \alpha_k & \beta_k \\ -\beta_k & \alpha_k \end{bmatrix} \tag{2.24}$$

By way of introduction transformations

$$\eta = (T^{-1} \otimes I_n)\xi \tag{2.25}$$

we derived that

$$\dot{\eta}_f = (I_N \otimes A + J \otimes BK)\eta \tag{2.26}$$

where $\eta = [\eta_f \, \eta^t_t]$ .Consequently, by tinkering with the composition of L , we obtained

$$\dot{\eta}_f = (I_M \otimes A + J_f \otimes BK)\eta_f \tag{2.27}$$

J_f which $f = 1, \ldots\ldots, M$, is derived from

$$J = \begin{bmatrix} J_f & 0_{Mx(N-M)} \\ 0_{(N-M)xM)} & 0_{(N-M)x(N-M)} \end{bmatrix} \tag{2.28}$$

2.6 Conclusion

In this chapter, various mathematical aspects of consensus control, such as graph theory and the Lyapunov stability theory, are broken down and explained in detail. In particular, the graph theory provided information regarding the communication topology among agents, whereas the Lyapunov stability theory provided steps for stability conditions that each agent should meet in order to be able to arrive at a consensus. Both of these theories were utilised in conjunction with one another. The knowledge that was required was provided by a combination of the two ideas. To put it another way, having access to these different kinds of information was essential in order to turn the concept of consensus control in multi-agent systems into a practical application. All of the fundamentals of linear state-feedback controller, observer, and observer-based controller design have been discussed up until this point. After that, a connection was made between them and the design of a consensus control for a multi-agent system, and a discussion was had regarding the conditions under which the systems are stable. As a result of it, a summary of some of the most significant discoveries made in the field of consensus control of multi-agent systems has been provided. In a nutshell, these may be placed in one of three categories: the first-order (single integrator), the second-order (double integrators), or the higher-order consensus. Explanations in greater detail were given for the characteristics that were shared by each group. In addition to this, it was said that the fundamental objective of this thesis is to create a higher-order consensus control design for multi-agent systems. This was discussed earlier.

CHAPTER 3

METHODOLOGY

3.1 Introduction

During the course of this chapter, the methodology of the project will be presented. This methodology will describe the work planning and flow of the project. In this chapter, we will also compare the results of using an observer for containment with and without a time delay, as well as using an observer-based controller with and without a time delay. Publications draw their conclusions on a comparison of the outcomes of using a consensus observer and an observer based controller. This comparison is based on the results of both approaches.

3.1.1 Project Flow

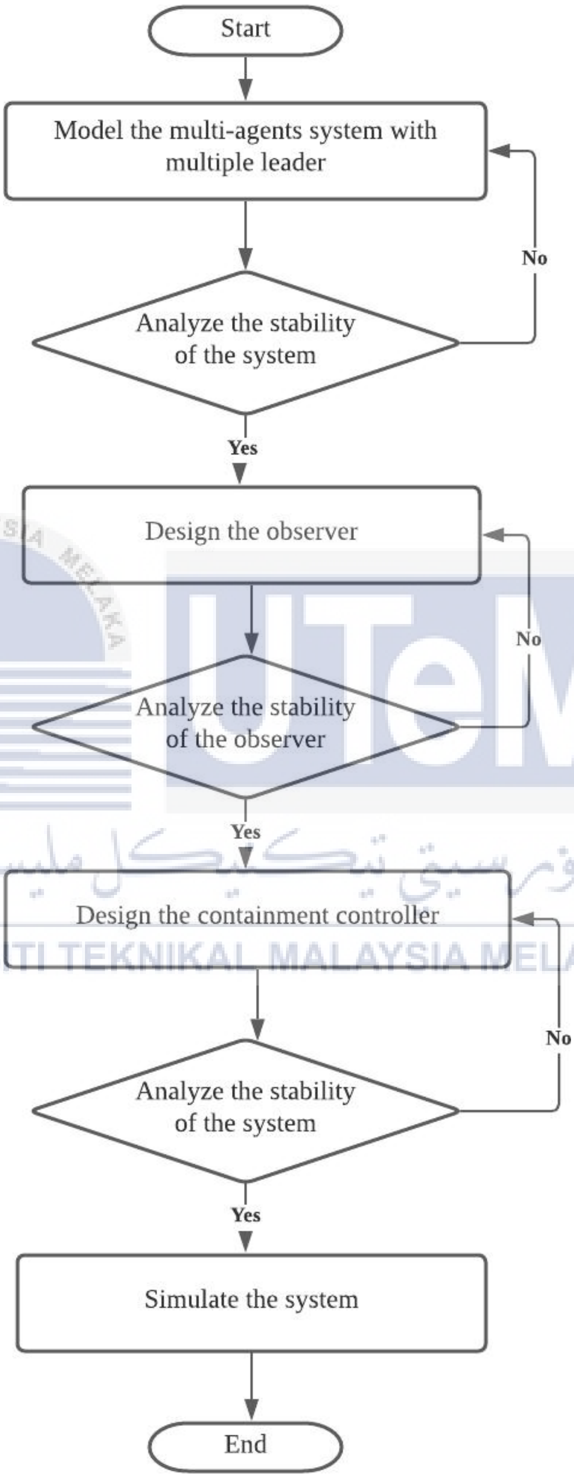

Figure 3.1: Project Flowchart.

3.1.2 Problem Statement

$N + 1$ nonlinear subsystems which are the same are told by the state space methodology:

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

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

Numerous connections and subsystems represented by a sequence of edges. A directed graph G identifies the connections between the various components of the system [26]. The nodes in this network are called vertices. V that represent the ξ. Graph G is linked to adjacency matrix A. $a_{ij} = 1$ if the subsystems j and I are coupled. $a_{ij} = 0$ if a_{ij} is not zero. From the adjacency matrix, we may derive the Laplacian matrix $L = l_{ij}$ as follows:

$$l_{ij} = -a_{ij} \, if \, j \neq i$$

$$l_{ii} = \sum_{j=i, \neq i}^{N} a_{ij} \tag{3.3}$$

For stability system, the eigenvalue must be less than 0.

$$eig(A - \lambda_i BK_c) < 0 \tag{3.4}$$

Then, laplacian matrix

$$L = \begin{matrix} 0 & 0_{1x} \\ l_1 & \bar{L} \end{matrix} \tag{3.5}$$

Observer for the state variables :

$$\dot{\hat{x}}_i = A\hat{x}_i + J(y_i - C\hat{x}) \tag{3.6}$$

If design an obsever for a system alone, it may look like this :

$$\dot{\hat{x}}_i = A\hat{x}_i + J\sum_{j=1}^{N} l_{ij}(y_j - C_{\hat{x}_j}) \tag{3.7}$$

Where J is the observer gain to make (A+JC) Hurwitz

We consider a controller [27], $U_i = -K\hat{x}_i$

To make $(A - BK)$ Hurwitz, you need a control gain of K. If we can identify a J suchthat the consensus control via dynamic output feedback is solved,

$$A\hat{x} - \lambda_i \bar{J}\bar{C} \tag{3.8}$$

Hurwitz for all the non zero eigenvalues of L.Take note of this :

$$A\hat{x} - \Lambda_i \bar{J}\bar{C} = \begin{matrix} A - BK & BK \\ 0 & A - \lambda_i JC \end{matrix} \tag{3.9}$$

As a result, the consensus control challenge is to choose an observer gain J such that $(A - \lambda_i JC)$ is Hurwitz. If (A,C) can be observed and G is tightly coupled, then a J like this must exist.

3.2 Containment Controller Design for Multi-Agents System (MAS) without time delay

In this section, explaination about the design of a containment controller for Multi-Agents System (MAS) without a non-linearity elements which is the time delay. Where the first step is to design the containment observer and then build the controller based on the containment observer.

Let

$$z(t) = x(t) + \int_t^{t+h} e^{A(t+\tau)} Bu \tag{3.10}$$

Differentiating $z(t)$ against time yield [28]

$$\dot{z}(t) = Ax(t) + e^{(-Ah)} Bu(t) + A \int_t^{t+h} e^{A(t+\tau)} Bu \tag{3.11}$$

$$= Az(t) + Du(t) \tag{3.12}$$

where $D = e^{Ah}B$ and the controller consider by

$$u(t) = Kz(t) \tag{3.13}$$

Consequently, applying the same technique (3.2), the equation of the consensus observer can be described as

$$\dot{\hat{z}}(t) = A\hat{z}_i(t) + Du_i + LC\sum_{j=1}^{N} l_{ij}\tilde{z}_j \tag{3.14}$$

where $\hat{z}_i \varepsilon R^n$ each subsystem is assessed to be in its current state $i = 1, \ldots, N$ and $L \varepsilon R^{nxq}$ is the observer's gain matrix.It can thus be deduced that estimating error dynamics

$$\dot{\hat{z}}(t) = A\hat{z}_i(t) - LC\sum_{j=1}^{N} l_{ij}\tilde{z}_j \tag{3.15}$$

where $\tilde{z}_i = z_i - \hat{z}_i$ with $\tilde{z} = [\tilde{z}_1^T, \ldots, \tilde{z}_1^T]^T$ For each subsystem, the dynamics of estimated error can be layered in a compact manner.

$$\dot{\tilde{z}}(t) = [I_N \otimes A - \bar{L} \otimes LC]\tilde{z} \tag{3.16}$$

Then came the introduction of state change.

$$\dot{\tilde{\eta}} = (T^{-1} \otimes I_n)\tilde{z} \tag{3.17}$$

with $T^{-1}\bar{L}T = \bar{J}$, the consensus observer's transformed dynamicscan be summed up as

$$\dot{\tilde{\eta}} = [I_N \otimes A - \bar{J} \otimes LC]\tilde{\eta} \tag{3.18}$$

where $\tilde{\eta} = [\bar{\eta}_1^T, \ldots, \bar{\eta}_N^T]^T$

3.2.1 Containment Observer Based Controller Design

The controller takes the equation (3.4) and the augmented closed-loop network dynam-icscompactly arranged in a stack [29]

$$\begin{bmatrix} \dot{\xi} \\ \dot{\tilde{z}} \end{bmatrix} = \begin{bmatrix} I_N \otimes (A - DK) & I_N \otimes DK \\ 0 & I_N \otimes A - L \otimes LC \end{bmatrix} \begin{bmatrix} \xi \\ \tilde{z} \end{bmatrix} \tag{3.19}$$

Using the same change, we were able to alter the dynamics of the network (3.8) for $\tilde{\eta}$

$$\begin{bmatrix} \dot{\xi} \\ \dot{\tilde{\eta}} \end{bmatrix} = \begin{bmatrix} I_N \otimes (A - DK) & I_N \otimes DK \\ 0 & I_N \otimes A - J \otimes LC \end{bmatrix} \begin{bmatrix} \xi \\ \tilde{\eta} \end{bmatrix} \tag{3.20}$$

3.3 Containment Controller Design for Multi-Agents System (MAS) with time delay

In this section, explaination about the design of a containment controller for Multi-Agents System (MAS) with a time delay. Where the first step is to design the contain-ment observer and then build the controller based on the containment observer. The final steps have been taken to ensure that the system is stable.

3.3.1 Containment Observer Design

Let

$$z(t) = x(t) + \int_{t}^{t+h} e^{A(t+\tau)} B u(\tau - h) d\tau \tag{3.21}$$

Differentiating $z(t)$ against time yield

$$\dot{z}(t) = Ax(t) + e^{(-Ah)}Bu(\,) + A\int_{t}^{t+h} e^{A(t+\tau)}Bu(-h)d\tau \qquad (3.22)$$

$$= Az(t) + Du(t) \qquad (3.23)$$

where $D = e^{Ah}B$ and the controller consider by

$$u(t) = Kz(t) \qquad (3.24)$$

So using the same approach (3.2),The consensus observer's equation is as follows:

$$\dot{\hat{z}}(t) = A\hat{z}_i(t) + Du_i + LC\sum_{j=1}^{N} l_{ij}\tilde{z}_j \qquad (3.25)$$

where $\hat{z}_i \varepsilon R^n$ is the estimated state of each subsystem for $i = 1, \ldots, N$ and $L \varepsilon R^{nxq}$ is the observer's gain matrix. It can thus be deduced that estimating error dynamics

$$\dot{\hat{z}}(t) = A\hat{z}_i(t) - LC\sum_{j=1}^{N} l_{ij}\tilde{z}_j \qquad (3.26)$$

where $\tilde{z}_i = z_i - \hat{z}_i$ with $\tilde{z} = [\tilde{z}_1^T, \ldots, \tilde{z}_1^T]^T$ Each subsystem's estimated error dynamics can be piled in a compact manner. [30].

$$\dot{\tilde{z}}(t) = [I_N \otimes A - \bar{L} \otimes LC]\tilde{z} \qquad (3.27)$$

Then came the introduction of state change.

$$\dot{\eta} = (T^{-1} \otimes I_n)\tilde{z} \qquad (3.28)$$

with $T^{-1}\bar{L}T = \bar{J}$, The dynamics of the consensus observer's altered dynamics can be characterised by this

$$\dot{\tilde{\eta}} = [I_N \otimes A - \bar{J} \otimes LC]\tilde{\eta} \tag{3.29}$$

where $\tilde{\eta} = [\bar{\eta}_1^T, \ldots.., \bar{\eta}_N^T]^T$

3.3.2 Containment Observer Based Controller Design

Closed-loop dynamics could be compactly stacked due to a controller .

$$\begin{bmatrix} \dot{\xi} \\ \dot{\tilde{z}} \end{bmatrix} = \begin{bmatrix} I_N \otimes (A - DK) & I_N \otimes DK \\ 0 & I_N \otimes A - L \otimes LC \end{bmatrix} \begin{bmatrix} \xi \\ \tilde{z} \end{bmatrix} \tag{3.30}$$

Using the same change, we were able to alter the dynamics of the network.(3.8) for $\tilde{\eta}$

$$\begin{bmatrix} \dot{\xi} \\ \dot{\tilde{\eta}} \end{bmatrix} = \begin{bmatrix} I_N \otimes (A - DK) & I_N \otimes DK \\ 0 & I_N \otimes A - J \otimes LC \end{bmatrix} \begin{bmatrix} \xi \\ \tilde{\eta} \end{bmatrix} \tag{3.31}$$

3.3.2.1 Stability Analysis

An observer for the state variables must be built for dynamic output feedback. Sub-system i information is comprised of its own output and the outputs of its adjacent subsystems. Subsystem i are the subsystems j with $q_{ij} = 1$. An observer for a single subsystem would look like this:

$$\dot{\hat{x}}_i = A\hat{x}_1 + J(y_i - C\hat{x}) \tag{3.32}$$

To make $(A + JC)$ Hurwitz, the observer gains J.

$$\dot{\hat{x}}_i = A\hat{x}_i + Bu_i + J\sum_{j=1}^{N} l_{ij}(y_j - C\hat{x}_j) \tag{3.33}$$

In this case, the observer gain matrix J will be determined at a subsequent time. Input to each subsystem's control system is provided by

$$u_i = -Kx_i \tag{3.34}$$

A $(A - BK)$ Hurwitz control gain is represented by K.It is therefore possible to writethe closed-loop control system.

$$\dot{x}_i = (A - BK)x_i + BK\tilde{x}_i \tag{3.35}$$

$$\dot{\tilde{x}}_i = A\tilde{x}_i + J\sum_{j=1}^{N} l_{ij}C\tilde{x}_j \tag{3.36}$$

Let $\bar{x}_i = [x_i^T, \tilde{x}_i^T]^T$, where $\tilde{x}_i = x_i - \hat{x}_i$ .We have

$$\dot{\bar{x}}_i = \bar{A}\bar{x}_i + \sum_{j=1}^{N} l_{ij}\bar{J}\bar{C}\bar{x}_j \tag{3.37}$$

where,

$$\bar{A} = \begin{bmatrix} A - BK & BK \\ 0 & A \end{bmatrix}, \tag{3.38}$$

$$\bar{J} = \begin{bmatrix} 0 \\ J \end{bmatrix}, \bar{C} = \begin{bmatrix} 0 & C \end{bmatrix} \tag{3.39}$$

In the same way, let us

$$\bar{x} = [\bar{x}_i^T, \ldots\ldots, \bar{x}_N^T]^T \tag{3.40}$$

and we have

$$\dot{\bar{x}} = (I \otimes \bar{A})\bar{x} + (L \otimes \bar{J}\bar{C})\bar{x} \tag{3.41}$$

If we can identify a J such that the consensus control through dynamic output feedback is solved,

$$\bar{A}\bar{x} - \lambda_i \bar{J}\bar{C} \tag{3.42}$$

Hurwitz for all nonzero eigenvalues of L. Take note of the fact that

$$\bar{A}\bar{x} - \lambda_i \bar{J}\bar{C} = \begin{matrix} A - BK & BK \\ 0 & A - \lambda_i JC \end{matrix} \tag{3.43}$$

Finding an observer gain J so that $(A - \lambda_i JC)$ Hurwitz becomes the consensus control problem.As long as (A, C) and G can be observed, there will always be a J.

CHAPTER 4

RESULTS AND ANALYSIS

4.1 Stabilization analysis

The eigenvalue needs to be less than 0 in order for there to be stability. According to the illustration that may be found in the Matlab below, the eigenvalue is less than 0. This demonstrates that the system does not exhibit any instability, the stability is archieved.

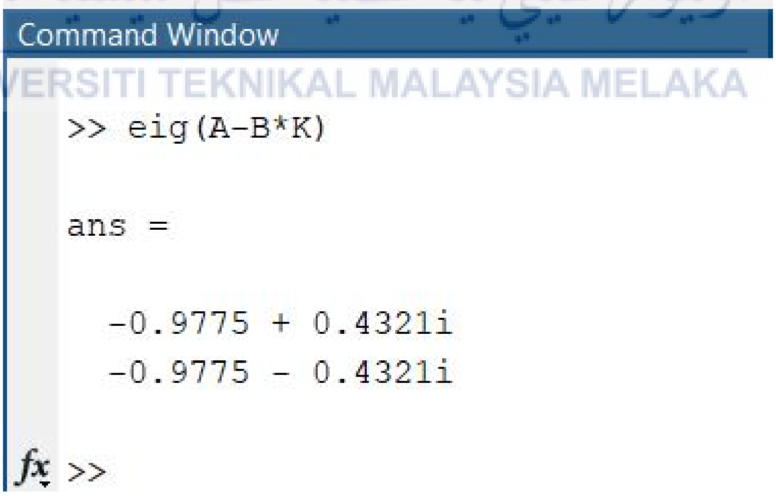

Figure 4.1: stabilization of Eigenvalues in Matlab.

From 4.1, show that the eigenvalues is -0.9975 where the value is less than 0 . Proved that the system is stable.

4.2 The Model of multi-agents system Graph connection and Laplacian Matrix

In this section, we constructed four different kinds of graph connections so that we could evaluate their performance.Then, we will discuss about the results that we obtain from each graph connection based on how well they work.

4.2.1 Directed Spanning tree

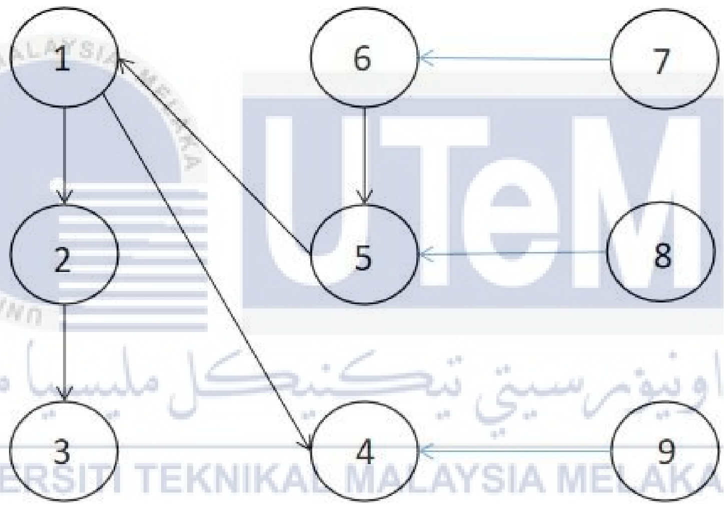

Figure 4.2: Graph connection of the Directed Spanning Tree.

Agents are connected to numerous leaders via a directed spanning tree. Since each node in the network receives information from the leader and then passes it on to the others, every node in the graph will have access to the leader's data [5]. That agent 6 received the data from leader 7 and subsequently passed it on to agent 5 is demonstrated. Repeatedly, the flow is passed to the other agents.

1	0	0	0	-1	0	0	0	0
-1	1	0	0	0	0	0	0	0
0	-1	1	0	0	0	0	0	0
-1	0	0	2	0	0	0	0	-1
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

Figure 4.3: Laplacian Matrix of the Directed Spanning Tree.

The laplacian's value can be determined by looking at the graph of connections. The graph's value can be determined by the direction in which each leader and agent are connected.

4.2.2 Undirected Spanning tree

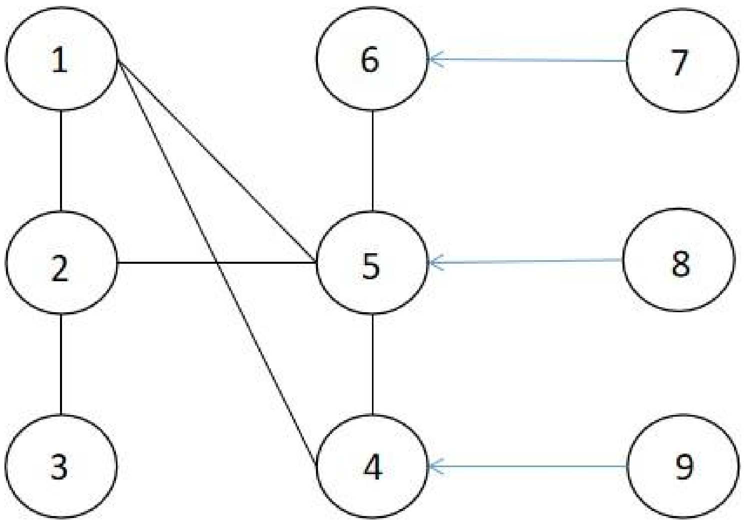

Figure 4.4: Graph connection of the Undirected Spanning Tree.

The undirected spanning tree connects the numerous leaders with their agents. The nodes in the graph are connected to one another randomly. Agents receive information from the leader and then pass it on to other agents, therefore every agent will receive the same information. An undirected system means that the agents can freely exchange information with each other without any certain direction.Repeatingly, the flow is passed on to the other employees.

3	-1	0	-1	-1	0	0	0	0
-1	3	-1	0	0	0	0	0	0
0	-1	1	0	0	0	0	0	0
-1	0	0	3	-1	0	0	0	-1
-1	-1	0	-1	5	-1	0	-1	0
0	0	0	0	-1	2	-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

Figure 4.5: Laplacian Matrix of the Undirected Spanning Tree.

The laplacian's value can be determined by looking at the graph of connections. The graph's value can be determined by the direction in which each leader and agent are connected.

4.2.3 Directed Graph with strongly connected

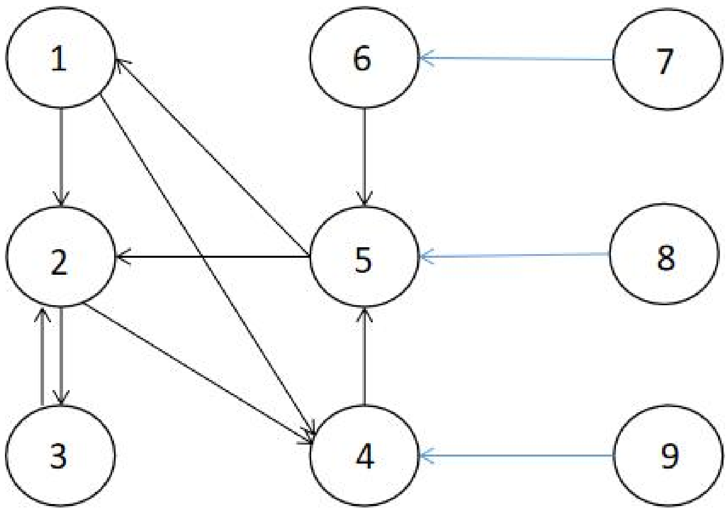

Figure 4.6: Graph connection with a Strongly Connected Agents

It's basically the same as the directed graph, except that the strongly connected indicates each agent's connection to the rest of the network is stronger. When the graph is well-connected, information is transmitted more quickly.

1	0	0	0	-1	0	0	0	0
-1	3	-1	0	-1	0	0	0	0
0	-1	1	0	0	0	0	0	0
-1	-1	0	3	0	0	0	0	-1
0	0	0	-1	3	-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

Figure 4.7: Laplacian Matrix with a Strongly Connected Agents.

The laplacian's value can be determined by looking at the graph of connections. The graph's value can be determined by the direction in which each leader and agent are connected.

4.2.4 Graph connection with a not Strongly Connected Agents

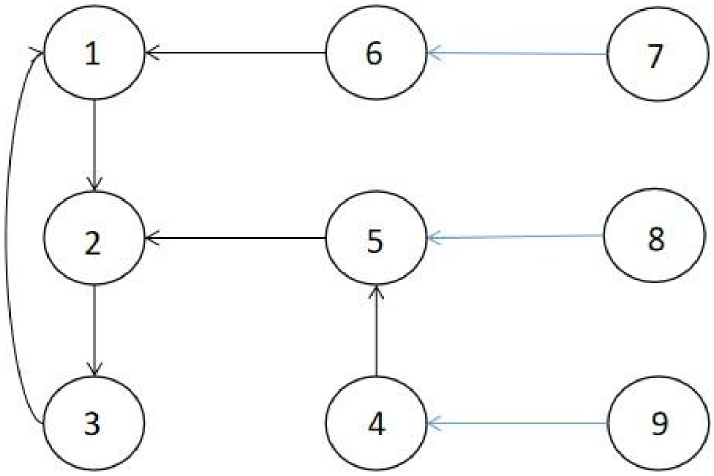

Figure 4.8: Graph connection with a not Strongly Connected Agents

This type of connection occurs in a graph when not all agents are connected to each other due to the lack of a Strongly Connected Agent. The agents can only communicate in one direction. As a result, the link between the agents is weak, and as a result, the connection is not to strong.

2	0	-1	0	0	-1	0	0	0
-1	2	0	0	-1	0	0	0	0
0	-1	1	0	0	0	0	0	0
0	0	0	1	0	0	0	0	-1
0	0	0	-1	2	0	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

Figure 4.9: Laplacian Matrix with a not Strongly Connected Agents.

The laplacian's value can be determined by looking at the graph of connections. The graph's value can be determined by the direction in which each leader and agent are connected.

4.2.5 Observer-Based Containment Controller for Multi-Agents System (MAS) without time delay

This is the graph result without the time delay. The graph result will display 2 graph because of the laplacian matrix is 2x2 so that it produce 2 data. The first graph is the plotted graph of leader & follower substance 1 for the point 1,3,5,7,... meanwhile the other one is plotted graph of leader & follower substancet for point 2,4,6,8,...

4.2.5.1 Directed Spanning Tree Connection

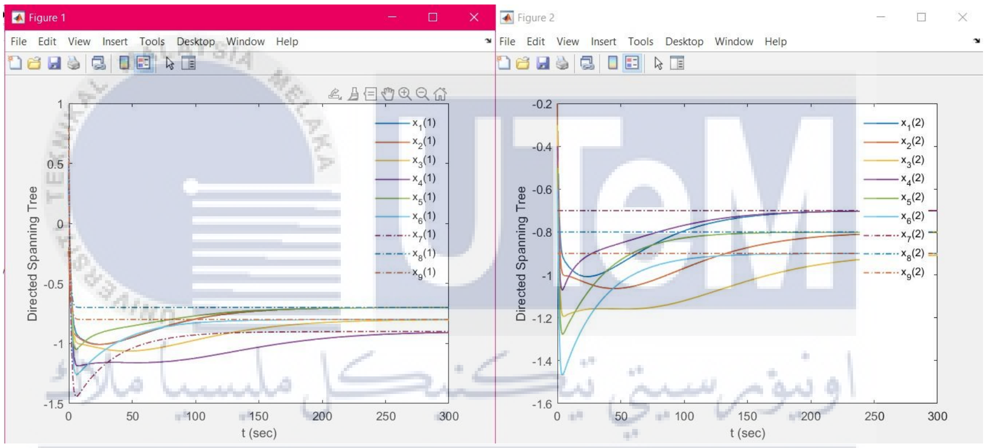

Figure 4.10: Directed Spanning Tree Connection without Time Delay.

4.2.5.2 Discussion

For the Directed Spanning Tree Connection without Time Delay. According to the graph result, the multi leader can keep the agents contain via a direct spanning tree connection with no time delay.As can be seen from the graph, the initial signal is dispersed, but after about 250 seconds, the agents begin to contain the leaders.

4.2.5.3 Undirected Spanning Tree Connection

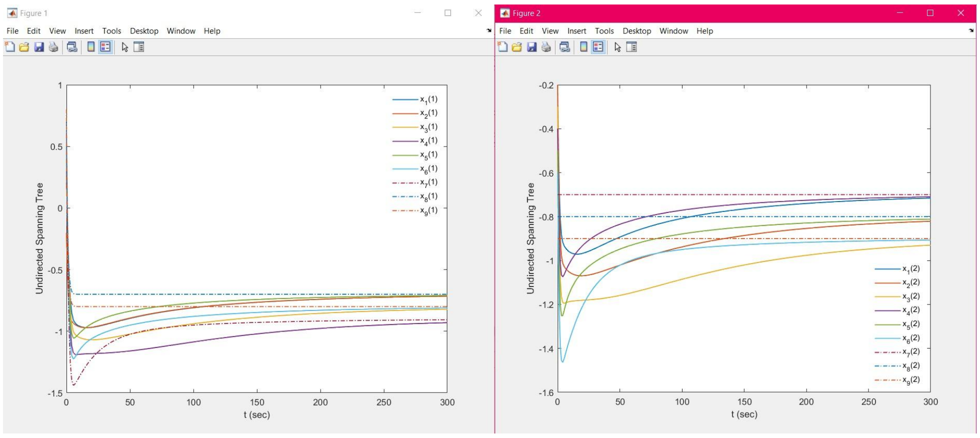

Figure 4.11: Undirected Spanning Tree Connection without Time Delay.

4.2.5.4 Discussion

For Undirected Spanning Tree Connection. According to the graph result, the multi leader can keep the agents contain via a direct spanning tree connection with no time delay.As can be seen from the graph, the initial signal is dispersed, but after about 250 seconds, the agents begin to contain the leaders.

4.2.5.5 Directed Graph with a Strongly Connected Agents

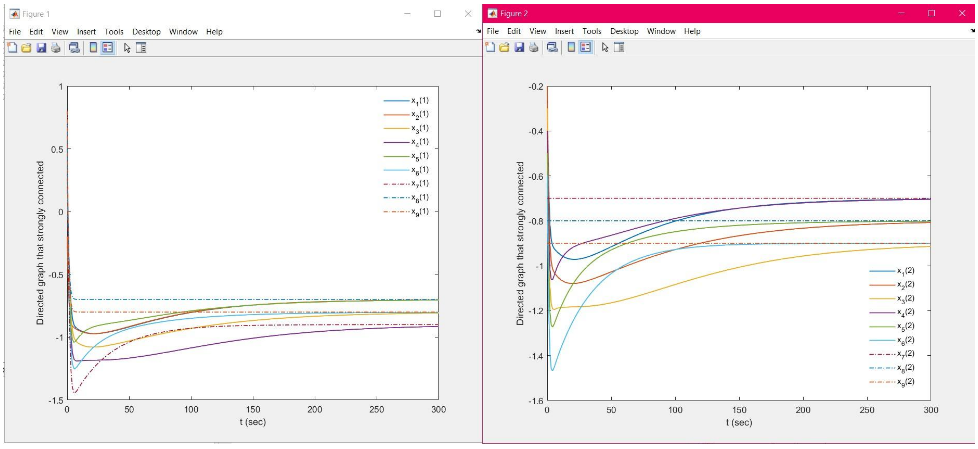

Figure 4.12: Directed Graph with a Strongly Connected Agents without Time Delay.

4.2.5.6 Discussion

For Directed Graph with a Strongly Connected Agents. According to the graph result, the multi leader can keep the agents contain via a direct spanning tree connection with no time delay.As can be seen from the graph, the initial signal is dispersed, but after about 250 seconds, the agents begin to contain the leaders.

4.2.5.7 Directed Graph with a Not Strongly Connected Agents

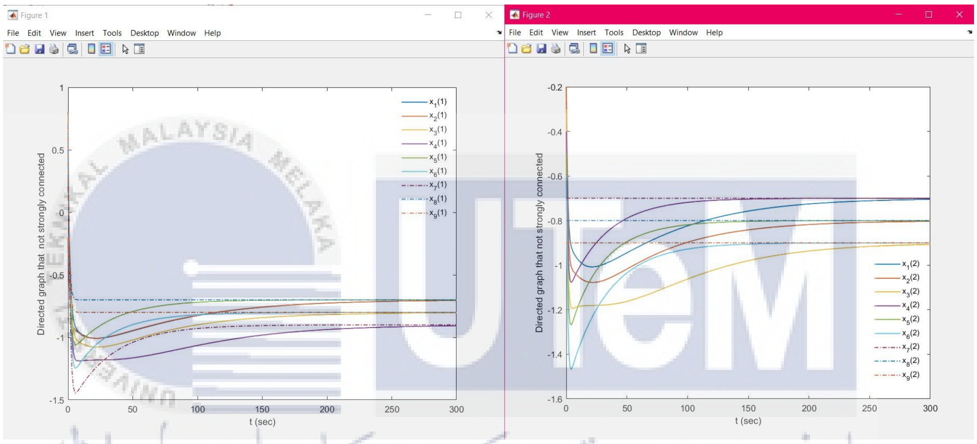

Figure 4.13: Directed Graph with a Not Strongly Connected Agents without Time Delay.

4.2.5.8 Discussion

For Directed Graph with a Not Strongly Connected Agents. According to the graph result, the multi leader can keep the agents contain via a direct spanning tree connection with no time delay.As can be seen from the graph, the initial signal is dispersed, but after about 250 seconds, the agents begin to contain the leaders.

4.2.6 Observer-Based Containment Controller for Multi-Agents System (MAS) with time delay

This is the graph result with the time delay. Here the time delay that has been set is 1.3 which is the max value of time delay that can still maintain the containment system. The graph result will display 2 graph because of the laplacian matrix is 2x2 so that it produce 2 data. The first graph is the plotted graph of leader & follower substance 1 for the point 1,3,5,7,... meanwhile the other one is plotted graph of leader & follower substance for point 2,4,6,8,...

4.2.6.1 Directed Spanning Tree Connection

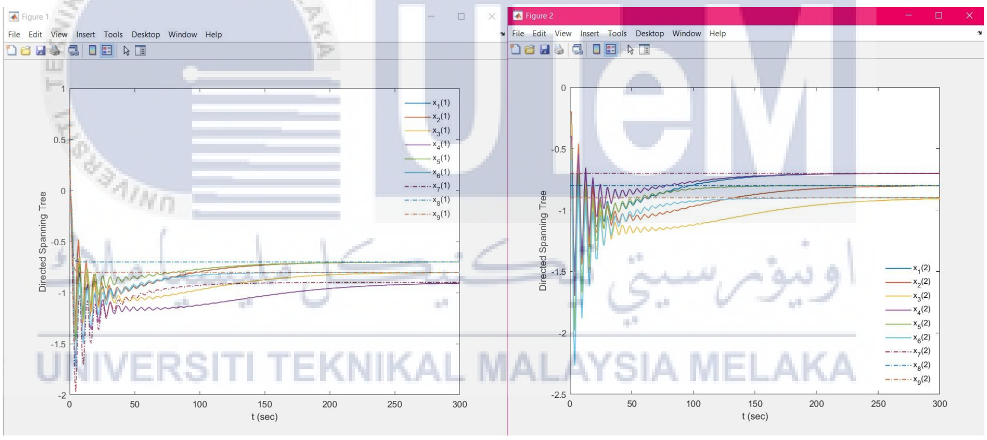

Figure 4.14: Directed Spanning Tree Connection.

4.2.6.2 Discussion

For Directed Spanning Tree Connection. According to the graph result, the multi leader can keep the agents contain via a direct spanning tree connection with time delay 1.3 .As can be seen from the graph, the initial signal is very wavy and scattered because of the time delay as a non-linearity elements, but after about 250 seconds, the agents begin to contain the leaders.

4.2.6.3 Undirected Spanning Tree Connection

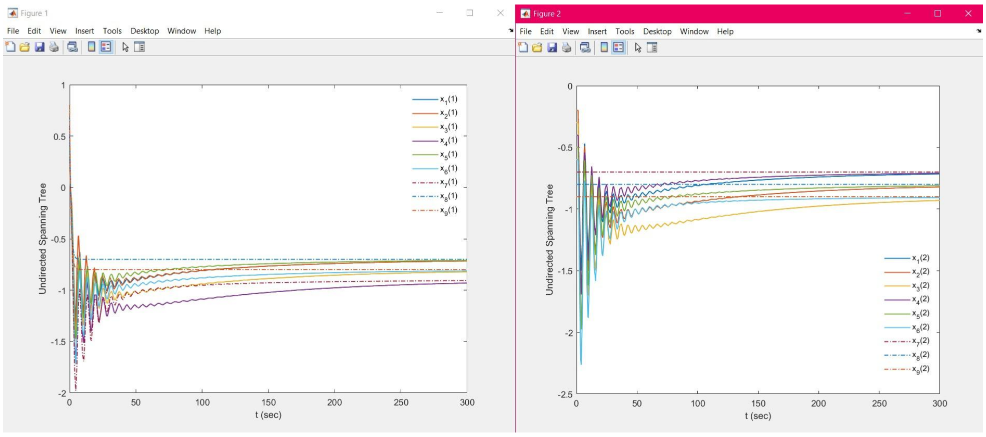

Figure 4.15: Undirected Spanning Tree Connection.

4.2.6.4 Discussion

According to the graph result, the multi leader can keep the agents contain via a direct spanning tree connection with time delay 1.3 .As can be seen from the graph, the initial signal is very wavy and scattered because of the time delay as a non-linearity elements, but after about 250 seconds, the agents begin to contain the leaders.

4.2.6.5 Directed Graph with a Strongly Connected Agents

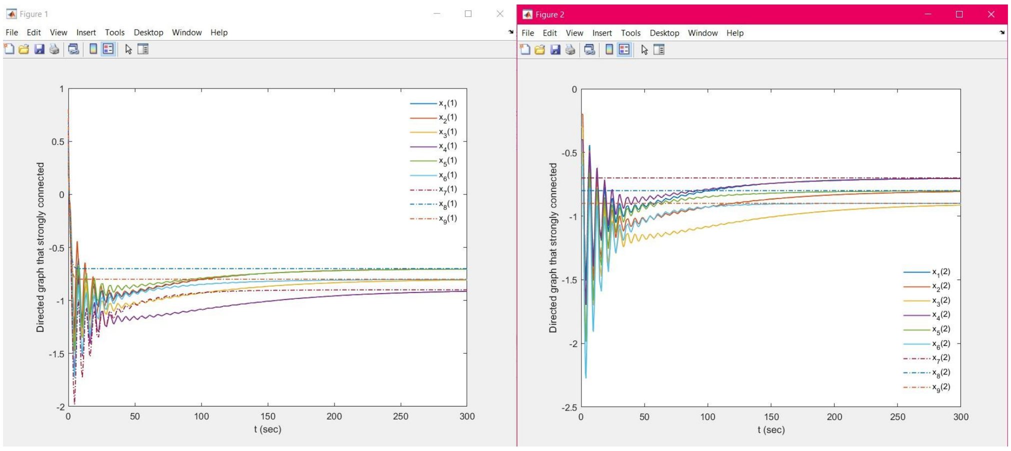

Figure 4.16: Directed Graph with a Strongly Connected Agents.

4.2.6.6 Discussion

According to the graph result, the multi leader can keep the agents contain via a direct spanning tree connection with time delay 1.3 .As can be seen from the graph, the initial signal is very wavy and scattered because of the time delay as a non-linearity elements, but after about 250 seconds, the agents begin to contain the leaders.

4.2.6.7 Directed Graph with a Not Strongly Connected Agents

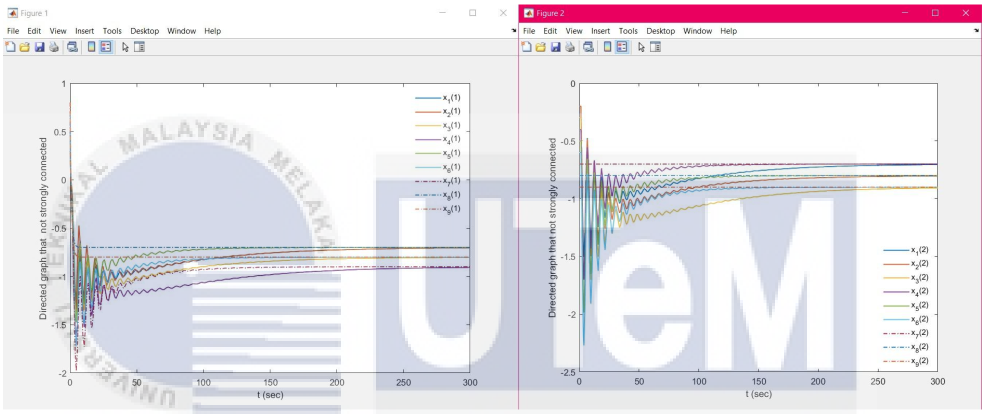

Figure 4.17: Directed Graph with a Not Strongly Connected Agents.

4.2.6.8 Discussion

According to the graph result, the multi leader can keep the agents contain via a direct spanning tree connection with time delay 1.3 .As can be seen from the graph, the initial signal is very wavy and scattered because of the time delay as a non-linearity elements, but after about 250 seconds, the agents begin to contain the leaders.

4.2.7 Comparison between Directed graph and Undirected graph

In this section, an overview of the comparison between directed and undirected graphs is presented.

Table 4.1: Comparison between Directed graph and Undirected graph

Graph connection	Summary
Directed Spanning Tree Connection	It's easier for the information to propagate since the connection between the leader and the agents is directed strongly because the one agents can sees the other agents in a time. Therefore, the agents are able to contain the leader more efficiently.
Undirected Spanning Tree Connection	They aren't as close as they could be since the connections aren't fully connected on one another. However, the information may still be obtained, but not fast as the directed spanning tree connection.

CHAPTER 5

DISCUSSIONS

5.1 Discussion

In order to determine the best connection for the graph pattern, we tested four different kinds of connections. It has been demonstrated that all connections are still able to reach the containment, but the directed spanning graph with strongly connected nodes produces the best results. This is due to the fact that the connection between each agent and the leader is robust due to the fact that the agents share data with one another, ensuring that no agent will experience any information loss from the leader. In contrast to the directed spanning graph, which does not strongly connect its nodes, this graph connection can still reach the containment. However, the information that is gained is obtained somewhat belatedly due to the fact that not all agents connect to each other.

According to all of the results, it has been demonstrated that the containment can be reached via each and every possible connection pattern in the graph. As long as there is a connection between the leader and the agents for the purpose of conveying the data and information, the graph connection can take on a variety of various patterns. In the case when there is no time delay, the initial waveform is smooth but a little bit scattered due to the absence of non-linearity elements. In contrast, when there is time delay, the initial waveform is smooth and consistent throughout.

In contrast to the time delay, the non-linearity parts of the time delay cause the earlier waveform to be more undulating and scattered, similar to the effect of noise interference.The input value for the amount of time delay has been checked in order to determine its maximum value. $t = 1.3$ is the highest number of time delay that can still reach the multiple leader of containment system, as determined by our experiment in Matlab, which led us to this conclusion. If the value is increased over that threshold, the waveform of the agents will become disorganised and will no longer be able to contain the leader. There isn't a significant difference in the amount of time it takes for the system to contain the leader regardless of whether it operates with or without a time delay. Nevertheless, when it operates with a time delay, there is a slight delay in reaching containment.

We came to the conclusion, based on all of these results, that the containment could still be reached, despite the delay in time. Simply by adding the delay in time, there is a limitation value that can be used to keep the containment system operational.But maintaining the system's stability is of the utmost importance without it, containment won't be possible, no matter what the other conditions are. Before beginning to perform the simulation, the eigenvalue needs to be checked in Matlab to ensure that its value is greater than zero. If it is not, the simulation will fail.

CHAPTER 6

CONCLUSION

6.1 Main Conclusion

In the course of this study, one of our key goals is going to be to provide information about containment management strategies that are predicated on output feedback for multi-agent systems that exhibit characteristics of non-linearity. For the purposes of this project, the target is a difficulty in the containment control of a multi-agent system when that system is contained by numerous leaders, and the ability of each agent to achieve a containment conclusion is the focus of the investigation. It is necessary for the leader of the containment control to take part inside the blockade maintained by the follower if they wish to prevent the system from becoming unstable.

The first objective of this project is to design a containment controller for a multi-agent system that also has non-linearity components. This objective was accomplished by modelling the system in Simulink by utilising the information flows that arise from the directed graph to construct the adjacency graph, which was then followed by the Laplacian graph.During the course of this inquiry, a suggestion was made regarding the application of an output-feedback containment controller to multi-agent systems that exhibited characteristics of nonlinearity.These networks have been directed to generate a topology similar to that of a forest.

The controller was successful in containing the states of the subsystems so that they were contained within the boundaries of the states of the leaders.This was accomplished through the application of a particular measure of nonlinear elements, which can be considered as being conservative but adequate for the system to accomplish its purpose of gaining containment. This was accomplished through the use of a specific measure of nonlinear components. In accordance with the findings of the Lyapunov stability analysis, the controller has been provided with adequate conditions to ensure its stability. It has been determined whether or not the non-linearity components affect the system's overall level of stability, and MATLAB's simulink performance has been utilised in order to mimic the behaviour of the system.

As a consequence of this, the controllers and observers have achieved the objective of their design, and as a consequence of this, they are now capable of containing a number of follower subsystems within a multi-agents system that has non-linearity elements.In the system with non-linearity and directed spanning forest topology network, the recommended containment controller with output feedback has been successful in enabling the leaders subsystems to restrict the follower subsystems. The model included a specific non-linearity measure that was thought to be conservative; nonetheless, the containment outcome was still able to be attained despite the inclusion of this measure. An in-depth investigation of the Laplacian structure as well as a Lyapunov stability analysis were able to identify the requirements that were necessary for the system to be stable. Running simulations, which verify that the containment attempt was effective, are used to validate the circumstances that have been established. Because of the hard work put in by everyone who was part in this endeavour, the objectives of this project have been realised with great success.

6.2 Future Recommendations

After the completion of this project, there are a few modifications that may be made to improve this containment control based on output input from the project dealing with multi-agent systems. These adjustments might be implemented to enhance the overall quality of the project. To begin, one conceivable option for improvement is to introduce extra non-linearity components, such as white noise and other components. This is only one of many possible ways to make improvements. This is one of the alternatives that could occur. In addition, able can incorporate various other kinds of controllers Next, an improvement that may be made to the containment system is to add additional leaders, such as 200 leaders, in order to expand the system so that it can support the larger project. This can be accomplished by expanding the number of leaders that are included in the system. In addition to that, the switching leader application is a future improvement that has the possibility of being included in this project. Because of this strategy for changing leaders, the current leader of the team can now switch places with one of the other agents to become the new leader while the team is still able to contain itself and advance in the same direction. In conclusion, but certainly not least, it is possible that this project could be improved by implementing it in a hardware system, which is an extremely uncommon event for this kind of system project in hardware.

REFERENCES

[1] L. Wang, T. Lei, J. Si, K. Xu, X. Wang, J. Wang, and Shouk, "Speed consensus control for a parallel six-wheel-legged robot on uneven terrain," *ISA Transactions*, 2021.

[2] A. S. M. Isira, Z. Zuo, and Z. Ding, "Leader-follower consensus control of lipschitz nonlinear systems by output feedback," *International Journal of Systems Science*, vol. 47, no. 16, pp. 3772–3781, 2016. [Online]. Available: https://doi.org/10.1080/00207721.2015.1122847

[3] D. Zheng, H. Zhang, and q. Zheng, "Consensus analysis of multi-agent systems under switching topologies by a topology-dependent average dwell time approach," *IET Control Theory & Applications*, vol. 11, 11 2016.

[4] L. Chen, C. Li, Y. Guo, G. Ma, Y. Li, and B. Xiao, "Formation–containment control of multi-agent systems with communication delays," *ISA Transactions*, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0019057821004961

[5] W. Ren and R. Beard, "Distributed consensus in multi-vehicle cooperative control: Theory and applications," *Communications and Control Engineering*, 01 2007.

[6] Z. Ding, "Consensus output regulation without state estimation for a class of nonlinear systems," in *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, 2012, pp. 5948–5953.

[7] MATLAB, *version 7.10.0 (R2010a)*. Natick, Massachusetts: The MathWorks Inc., 2010.

[8] R. Rajamani and Y. M. Cho, "Existence and design of observers for nonlinear systems: Relation to distance to unobservability," *International Journal of Control*, vol. 69, no. 5, pp. 717–731, 1998. [Online]. Available: https://doi.org/10.1080/002071798222640

[9] X. Chen and L. Zhao, "Observer-based finite-time attitude containment control of multiple spacecraft systems," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 68, no. 4, pp. 1273–1277, 2021.

[10] Y. Li, F. Qu, and S. Tong, "Observer-based fuzzy adaptive finite-time containment control of nonlinear multiagent systems with input delay," *IEEE Transactions on Cybernetics*, vol. 51, no. 1, pp. 126–137, 2021.

[11] J. Sheng, C. Shao, Z. Ding, and H. Ji, "Consensus control of a class of nonlinear systems with communication time delay," in *2014 American Control Conference*, 2014, pp. 819–824.

[12] X.-H. Wang and H.-B. Ji, "Leader-follower consensus for a class of nonlinear multi-agent systems," *International Journal of Control, Automation and Systems*, vol. 10, 02 2012.

[13] H. Li, "Chapter 6 - network topology design," in *Communications for Control in Cyber Physical Systems*, H. Li, Ed. Boston: Morgan Kaufmann, 2016, pp. 149–180. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780128019504000068

[14] L. Zou, Z. Wang, H. Gao, and F. E. Alsaadi, "Finite-horizon H_{∞} consensus control of time-varying multiagent systems with stochastic communication protocol," *IEEE Transactions on Cybernetics*, vol. 47, no. 8, pp. 1830–1840, Aug 2017.

[15] R. Faudree, "Graph theory," in *Encyclopedia of Physical Science and Technology (Third Edition)*, third edition ed., R. A. Meyers, Ed. New York: Academic Press, 2003, pp. 15–31. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B0122274105002969

[16] Z. Li, Z. Duan, W. Ren, and G. Feng, "Containment control of linear multi-agent systems with multiple leaders of bounded inputs using distributed continuous controllers," *International Journal of Robust and Nonlinear Control*, vol. 25, 05 2014.

[17] V. Fionda and L. Palopoli, "Biological network querying techniques: Analysis and comparison," *Journal of computational biology : a journal of computational molecular cell biology*, vol. 18, pp. 595–625, 03 2011.

[18] A. Y. Kibangou, "Graph laplacian based matrix design for finite-time distributed average consensus," in *2012 American Control Conference (ACC)*, 2012, pp. 1901–1906.

[19] J.-M. Réveillac, "2 - elements of graph theory," in *Optimization Tools for Logistics*, J.-M. Réveillac, Ed. Elsevier, 2015, pp. 11–29. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9781785480492500025

[20] A. Jamali, "Study on eigen values and eigen vectors of matrices: An iterative approach, ganit j. bangladesh math. soc. 37(2017) 39-50. (issn print. 1606-3694 ; online 2224-5111)," 12 2017.

[21] R. Varga, *Geršgorin and His Circles*, 01 2004, vol. 36.

[22] M. Attenborough, "19 - graph theory," in *Mathematics for Electrical Engineering and Computing*, M. Attenborough, Ed. Oxford: Newnes, 2003, pp. 461–478. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780750658553500450

[23] R. Sawada, Y. Sakumoto, C. Takano, and M. Aida, "Experimental study on relationship between indices of network structure and spectral distribution of graphs," 07 2016.

[24] "Chapter 1 - basic concepts," in *Adaptive Sliding Mode Neural Network Control for Nonlinear Systems*, ser. Emerging Methodologies and Applications in Modelling, Y. Li, J. Zhang, and Q. Wu, Eds. Academic Press, 2019, pp. 1–16. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B978012815372700001X

[25] S. Saat, S. K. Nguang, and A. Nasiri, "Chapter 1 - introduction," in *Analysis and Synthesis of Polynomial Discrete-Time Systems*, S. Saat, S. K. Nguang, and A. Nasiri, Eds. Butterworth-Heinemann, 2017, pp. 1–27. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780081019016000013

[26] S. N. S. Hanis, A. S. M. Isira, A. M. Yusop, M. H. Hairi, W. Chunyan, C. D. Casuat, and Z. A. Mutalip, "State feedback containment control of multi-agents system with lipschitz nonlinearity," vol. 23, no. NO 3.

[27] 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.

[28] S. N. S. Hanis, A. S. M. Isira, A. M. Yusop, C. Wang, M. H. Hairi, and S. Idris, "Observer based containment control for multi-agents system with lipchitz and input delay," in *2020 IEEE 8th Conference on Systems, Process and Control (ICSPC)*, 2020, pp. 74–79.

[29] D. Luenberger, "An introduction to observers," *IEEE Transactions on Automatic Control*, vol. 16, no. 6, pp. 596–602, 1971.

[30] P. Pagilla and Y. Zhu, "Controller and observer design for lipschitz nonlinear systems," in *Proceedings of the 2004 American Control Conference*, vol. 3, 2004, pp. 2379–2384 vol.3.