CHARACTERIZATION OF COOPERATIVE CONTROL FOR MULTIPLE NON-HOLONOMIC WHEELED MOBILE ROBOTS TO ACHIEVE FORMATION TRACKING

LEE ZHI XIAN

UNIVERSITI SAINS MALAYSIA

2017

CHARACTERIZATION OF COOPERATIVE CONTROL FOR MULTIPLE NON-HOLONOMIC WHEELED MOBILE ROBOTS TO ACHIEVE FORMATION TRACKING

by

LEE ZHI XIAN

A dissertation submitted for partial fulfillment of the requirement for the degree of Master of Science (Electronic System Design Engineering)

JUNE 2017

Acknowledgement

First of foremost, I would like to express my sincere gratitude to my research supervisor, Dr. Muhammad Nasiruddin bin Mahyuddin for giving me this opportunity to conduct the research under his supervision. His guidance and patience has helped me throughout the research period. I am grateful that he could share his knowledge especially with his immense knowledge in distributed control field of study. His feedback and idea help to make sure the research is progressing towards the goal. Besides, I would also thank him for spending his precious hour to help review the thesis and make sure it is well written.

I would also like to take this chance to thank the reviewers for their time in reviewing the thesis and ensure high quality thesis. Their comments and feedbacks are precious to complete this thesis.

Next, I would like to take this opportunity to thank the company I am working on, NI Malaysia for sponsoring the education assistance. This MSc degree is not possible without the sponsorship from NI Malaysia. I am grateful to have the support from the company and colleagues in pursuing this MSc degree.

Lastly, I would like to offer my deepest gratitude and blessing to my beloved family who have supported me throughout the project and my life in general. Their moral support and motivation was the key to keep me moving forward and achieving success in life.

Table of Contents

List of Figures

List of Tables

List of Symbols

l_{ij}	Relative distance
φ_{ij}	Relative angle
G	Graph set
V	Vertex set
E	Edge set
N	Number of agents
A	Adjacency matrix
a_{ij}	Adjacency matrix element
D	In-degree matrix
d_i	In-degree matrix element
L	Laplacian matrix
X_I	X axis of global reference frame
Y_I	Y axis of global reference frame
x_i	X position of robot in global reference frame
y_i	Y position of robot in global reference frame
θ_i	Heading of robot in global reference frame
v_i	Linear velocity in global reference frame
ω_i	Angular velocity in global reference frame
q_i	State vector
u_i	Control input
X_R	X axis of inertial coordinate frame
Y_R	Y axis of inertial coordinate frame
$R(\theta_i)$	Rotational matrix
$R^T(\theta_i)$	Transposed rotational matrix
$\mathcal{P}$	Geometric pattern

P_x	X axis in orthogonal coordinate system
P_y	Y axis in orthogonal coordinate system
p_{xi}	X position in orthogonal coordinate system
p_{yi}	Y position in orthogonal coordinate system
p_{x0}, p_{y0}	Origin of orthogonal coordinate system
z_{1i}	Transformed state 1
z_{2i}	Transformed state 2
z_{3i}	Transformed state 3
z_{10}	Virtual leader transformed state 1
z_{20}	Virtual leader transformed state 2
z_{30}	Virtual leader transformed state 3
B	Leader pinning matrix
b_i	Leader pinning matrix element
ζ_{1i}	Observer term for ω_i
ζ_{10}	Virtual leader ζ_1
k_ω	Gain constant
μ_i	Leader pinning strength
ζ_{2i}	Observer term for v_i
ζ_{20}	Virtual leader ζ_2
k_ζ	Gain constant
ρ_i	Gain constant
β_i	Gain constant
k_1	Gain constant
k_2	Gain constant
k_v	Gain constant
v_0	Virtual leader linear velocity
ω_0	Virtual leader angular velocity

Pencirian Kawalan Teragih Kerjasama Bagi Sekumpulan Robot Beroda Dalam Formasi Penjejakan

Abstrak

Kawalan teragih untuk sistem kerjasama adalah satu bidang penyelidikan yang baru dalam sistem kawalan. Penyelidikan ini tumpu dalam pencirian algoritma kawalan teragih kerjasam untuk menyelesaikan formasi penjejakan bagi sekumpulan robot beroda. Penyelidikan sedia ada tumpu dalam cara Matematik untuk bukti penumpuan kawalan teragih tetapi tiada yang menyelidik bagaimana parameter dalam algoritma menjejaskan kawalan. Selain itu, biasanya hanya satu topologi komunikasi diselidik dalam menyelesaikan formasi penjejakan. Jadi, penyelidikan ini akan mengisi jurang dengan pencirian parameter kawalan teragih kerjasama. Di samping itu, beberapa topologi komunikasi juga akan digunakan untuk menguji bagaimana cara ejen jiran komunikasi akan menjejaskan prestasi. Empat agen robot beroda akan diuji melalui simulasi. LabVIEW^{TM} digunakan untuk menjalankan simulasi tersebut. Penyelidikan ini telah menguji kawalan teragih kerjasama melalui pencirian parameter and juga beberapa topologi komunikasi. Selain itu, pencirian in mampu melengkapkan analisis Lyapunov sedia ada, dan membantu dalam kajian kawalan kerjasama robot beroda. Pencirian ini membantu untuk memahami setiap parameter dalam kawalan teragih dan melaraskan parameter untuk menyelesaikan isu formasi penjejakan. Sebahagian formasi penjejakan telah dicapai manakala kawalan teragih yang digunakan dalam penyelidikan ini masih boleh ditambah suai dengan menggunakan kawalan penyesuaian dalam algoritma kawalan teragih.

Characterization Of Cooperative Control For Multiple Non-holonomic Wheeled Mobile Robots To Achieve Formation Tracking

Abstract

Distributed control for cooperative system is an emerging research field in control system. This research focuses on characterization of the distributed control algorithm in solving formation tracking for multiple non-holonomic wheeled mobile robots. The existing research work mostly used mathematical approach to proof the convergence of controller but does not investigate how the parameters would affect the controller. Besides, usually only one communication topology is presented in solving formation tracking. Therefore, this research aims to fill in the gap on current research by performing characterization of gain parameters in the distributed controller studied. Besides, several communication topologies are evaluated to understand how the neighbouring agents connected will impact the performance. A multi-agent system that consists of four wheeled mobile robots are investigated in this research using simulation approach. LabVIEW^{TM} is used to simulate the multi-agent formation control. This research managed to perform the characterization of gain parameters and evaluate different communication topologies. The characterization would complement the existing Lyapunov analysis thereby improving the research in cooperative formation control of wheeled mobile robot. This has helped to understand the how the distributed controller studied and used to tune the controller to solve the formation tracking. The formation tracking control is partially achieved and can be further improved by making the parameters adaptive to achieve state consensus.

Chapter 1

Introduction

1.1 Background

Multi-agent cooperative system has gained the attention of researchers lately. The main goal of having multi-agent system is to overcome the limitations of individual agents. Solving difficult tasks with only single agent requires it to be very robust and powerful, usually that comes with trade-off such as complexity and cost. With the advancement in technology area such as sensor network and communication, it is preferable to have multiple agents to collectively achieve a common goal, due to simplicity per agent and offers more robustness as whole. Information sharing among agents plays an important role to achieve common goal as a system [2].

Besides, multi-agent cooperative system can be further divided into centralized and distributed scheme. The former scheme requires a centralized station that collects information from individual agents, processes and sends commands to agents. In this case, agents are only in charge of sensing and response. This scheme requires high communication bandwidth as the number of agents increases and limited by the communication range to central station. On the other hand, distributed cooperative approach overcomes the centralized scheme and offers a more robust scheme. It does not require a central machine for decision making, instead all the agents in the system are able to compute based on the neighbouring information available and generate their own commands.

Some of the control tasks that are being research to be solved using the multi-agent cooperative system includes rendezvous, formation control [3], flocking, and attitude alignment [4]. The applications of cooperative system includes formation control of Unmanned Aerial Vehicle (UAV), multi-vehicle rendezvous, micro-grids [5] and etc. With the vast application of cooperative control, the motivation of research

in this field is to create distributed consensus algorithm that will be applied onto individual agents. This algorithm uses the available information to compute and generate control to achieve consensus.

1.2 Problem Statements

Most graph theory results related to cooperative control are obtained for linear agents. However, many practical applications employing cooperative control involve nonlinear agent dynamics and need to consider nonholonomic control involving multiple nonholonomic and nonlinear agents are new to date.

Other common approach such that of [6, 7, 8] considered cooperative control for a case of specialised class of robotic system. The work in [6, 7, 8] achieves control objective for a subset of agents and not for the entire agents in a network.

Recent development by Tan et al [9] and Dong [3] revealed a cooperative control which suits formation control and leader following applications. Tan exploits the use of adaptive law to adjust the strength of the consensus control. However, convergence and stability analysis is yet to furnish.

None of the work had explored on in depth investigation to characterise the effects of each of the tuning parameters on the performance of the consensus formation tracking. Moreover, investigation on the subsequent effects of communication topologies on the consensus performance has not been thoroughly embarked.

1.3 Research Objectives

The objectives of this research are as below:

1. To characterize the effect of parameters in distributed consensus algorithm in solving formation tracking of multi-agent wheeled mobile robots

2. To evaluate the distributed consensus algorithm on different communication topologies in a multi-agent systems

1.4 Research Scope

This research will be focusing on characterization of distributed cooperative control of multi-agents non-holonomic wheeled mobile robot agents in solving formation tracking. In this case, the multi-agent refers to a system of four mobile robots will be investigated. In formation tracking, the multi-agents will perform the formation control and virtual leader containment tracking.

To evaluate the effect of communication topologies, several topologies are proposed. However, this research will only focus on a fixed topology on each simulation, which also means that the time varying effect of communication topologies are excluded. The time delay of information flow is also assumed to be negligible.

In this research, virtual leader is involved for the formation tracking. Therefore, there will be one virtual leader established. The virtual leader is assumed to be fixed throughout the simulation instead of varying agents switching roles as the virtual leader. While investigating, the formation tracking of multi-agent system, a desired trajectory is required on the virtual leader for the other agents to track. A circular trajectory is defined on the virtual leader, so the other agents will also require to follow a circular trajectory and containing the virtual leader.

Since this research is focusing on non-holonomic wheeled mobile robots, a simulation model is required to implement distributed consensus algorithm. Only the kinematics of wheeled mobile robot is considered, whereas the dynamics will be excluded. This is to reduce the complexity of consensus algorithm from all the non-linearity effects.

1.5 Research Contribution

The contribution of this research is to characterize the gain parameters in distributed control algorithm and its effect on the formation tracking of a virtual leader. Using the characterization information, the gain parameters are also adjusted to solve the formation tracking of four agents wheeled mobile robots. Besides, several communi-

cation topologies on information exchange are also investigated to understand how it impacts on the overall control objective. Lastly, this research also explores on using LabVIEW^{TM} to solve academia research problem with the help of control and simulation toolkit.

1.6 Dissertation Outline

This dissertation consists of five chapters. Chapter 1 briefly explains on the project details and scope. Chapter 2 presents the literature review on distributed control and some preliminary knowledge for this project. Chapter 3 presents the methodology applied to carry out this research. Chapter 4 depicts the numerical simulation results obtained from LabVIEW simulation. Lastly, Chapter 5 summarizes the findings of this dissertation.

Chapter 2

Literature Review

2.1 Introduction

This chapter will present the literature review of this research and provide some background knowledge of this research. In Section 2.2, some of the overview research done on the cooperative control area is presented together. In Section sec: formation control reviews some of the cooperative control research in solving formation control. In Section 2.4 presents some of the literature work in solving leader following problem. In section 2.5 discusses several containment control problem while Section 2.6 focuses on research with rendezvous problem. In Section 2.7, background knowledge on graph theory which as an essential mathematical tool for multi-agent system is explained.

2.2 Research background on Distributed Cooperative Control

Distributed cooperative control is a field in control theory that has caught the attention of researchers for the past decade. Distributed cooperative control approach has many advantages [10, 11] compared to centralised control approach, especially less complexity and system requirement. Key advantages such as higher robustness, flexibility, scalability, lower cost and agility in sensing and control can be gauged from the cooperative scheme. Distributed cooperative control has been implemented in various multi-agent applications such as in wheeled mobile robots [12], unmanned aerial vehicle [13, 14], underactuated surface vessels [15] and etc.

Huang et. al [10] has reviewed the recent progress in the field of distributed high order multi-agent coordination, as most of the engineering system are actually of

high order dynamics. In [10], various consensus algorithm presented are related to solving high order system. The consensus algorithm for both linear and non-linear system are discussed together with the limitation and challenges faced in each area of research.

Cao et. al [2] reviewed the progress in distributed cooperative control from the control problem perspective. The control problems are categorized into consensus, distributed formation, distributed optimization and distributed estimation. This paper introduces each control problem with the respective consensus protocol involved and highlights current research progress with the challenges ahead. Some of the challenges in actual application includes the delay effects due to limited communication bandwidth and the transmission and processing overhead time.

2.3 Formation control

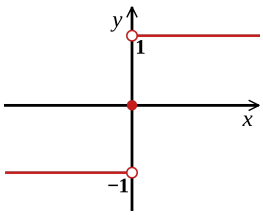
The goal in a formation control problem is the convergence of multiple agents into a formation from initial random position [16]. In his paper, Murray described formation control as the ability of multi-agent systems to maintain their position relative to each other [17]. If the final formation of the agents is required to meet a predefined geometric formation, this task is also known as formation stabilization.

Instead of using the direct information from local coordinate of nonholonomic robots, the local coordinate is transformed into chained form system. Similar state transformation method is also used in [3, 18, 19] in solving formation control of multi-agent robots. This state transformation also includes the information of desired geometric formation from the orthogonal coordinates, which can be used in the distributed control laws. For simulation purpose, this approach only considers the kinematics of robot agent to prove the controller design, whereas the real-world robot involves more dynamics to have an accurate model. In the paper presented, Lyapunov function is used to determine the convergence of the controller. However, the mathematical analysis approach does not indicate how each of the parameter used in the controller would affect the overall performance.

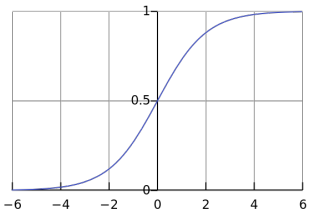
Based on Tan et. al [9] research, a distributed formation controller is proposed for a group of four generic agent system. In addition to maintaining desired formation, the distributed controller also allows the group of agents to perform virtual leader tracking. Adaptive control law is applied into the distributed controller to adjust the strategy dynamically. The intention of introducing adaptive control is to ensure the state of agent converges exponentially towards the desired trajectory. On the other hand, the convergence analysis of such proposed adaptive law have yet to be proven.

Formation control of multiple wheeled robot using the $l - \varphi$ control strategy is presented by Li et. al [20]. In this control method, the formation of multi-robot is done in leader-following structure where the relative distance, l_{ij} and relative angle φ_{ij} are used in the formulation. This approach is assumed that the relative distance and relative angle is available for formulation, where visual camera is used to determine this information. The limitation of this strategy is that it requires a method to track the error between robots, often done by a central machine before it could be used for distributed control on agent side.

Liao et. al [21] took a different approach in solving the formation control of multirobot dynamic system by using the distributed sliding mode controller. In his paper, distributed sliding mode formation control (DSMFC) is proposed to solve the formation control of dynamic systems with external disturbances and parameter uncertainty. The control algorithm is simulated on wheeled mobile robots model using the kinematics and dynamics. In the simulation, the signum function (Figure 2.1a) used in sliding mode control is replaced by saturation function (Figure 2.1b) to avoid singularity numerical computation as well as to soften the aggressive nature of the switching function. The closed loop system robustness stability is guaranteed by proving using Lyapunov Theorem.



(a) Signum function



(b) Saturation function

Figure 2.1: Sliding mode control function

2.4 Leader follower problem

Leader following consensus refers to convergence of follower agent to the leader agent to achieve the agreed state of the leader. Leader following is a fundamental problem in cooperative control since most of the cooperative control problem usually has a leader that determines the allows the follow to track and perform. In Xu et. al [22] research, investigation is done on a single leader, single follower configuration to understand how the leader dynamic uncertainty would affect the consensus problem. This paper introduces uncertainties and bounded external disturbances to both leader and follower, and adaptive control is used to reject the external disturbance introduced. This paper highlights the fundamental problem in disturbance that occurred in single leader, single follower scenario, but lacks information on how this can be further applied into multi-agent system.

In Shao. et al [23] research work, a framework is presented to use leader following approach in solving the formation control problem. The proposed solution involves three level hybrid control architecture, where the first level assigns a robot as the leader for trajectory planning. The second level is used to create leader-follower pair in a multi-agent system, where N robots are decomposed to $N-1$ leader-follower pair for tracking. The third level control is used to determine the role of agent either as leader or follower. This approach can be used to create multiple formations such as wedge-like formation, hexagon and column formation. However, the approach presented requires all the agents to form leader-following pair, which also meant only direct spanning tree topology is supported. This controller was not able to

utilize more than one leader agent information to generate stronger control inputs.

Bravo et. al [24] explored the formation control using leader-following approach for a group of three mobile robots. This work uses the switching control model that allows any robots in the system to be either leader or follower. The approach uses two level control stages, where the first stage is used to generate control trajectories and the second stage is based on cooperative control trajectory tracking control. The switching control implies on the role of agent, the leader agent will operate with the first stage control while the follower agent operates in second stage for trajectory tracking. The communication topology used for three agent system in research was the fixed, strongly connected graph whereby all agents have the information of the other neighbouring agents. Therefore, this leader-follower approach does not explain the effect of varying communication topology to other than strongly connected and how it would affect the overall control strategy.

Based on Ding. et. al [25] research, the effect of time-varying delays in decentralized formation of multiple UAVs is investigated. This research uses the leader-following consensus approach in solving formation control of UAVs. Lyapunov function is employed with the leader-follower consensus protocol to verify the performance of control strategy. The control strategy guaranteed by Lyapunov function could solve the formation control of UAVs for a fixed communication topology. On the other hand, his work does not clearly mention how different communication topology would affect the control problem.

2.5 Containment control

In a containment control problem, the system consists of multiple leader agents that formed a geometric space, where the follower agents are to be contained within. Ma. et. al [26] paper investigated on distributed containment control of multiple agents with unknown parameters. The research investigated the containment control of four leaders and six followers system. The control objectives are guaranteed for regulation case (fixed leaders) and dynamics tracking case (dynamic leaders) using

Lyapunov candidate function method. However, the research does not cover on a change in communication topology or missing information from one of the leader to the impact of overall control objective.

For Chen et. al [27], formation containment control for multiple Euler-Lagrange system with uncertainties and disturbances were studied. In this paper, an adaptive formation control algorithm using neural network was proposed for the leader system while the agent uses a separate distributed control law for track the leader. Since the approach used requires separate controller for leader and agent, this also means that it is not possible to switch the role without changing controller. The controller used was not generic such that it is applicable to both leader and follower of the system.

In Di et. al [28] paper, finite time containment control of multi-agent network was investigated with the nonlinear dynamics. Two topologies are studied, which are the undirected connected leader-follower topology and the strong connected leader-follower topology. Variable structure control was use in the first topology whereas non-singular fast terminal sliding mode surface was used for the second. The controller was guaranteed using Lyapunov function but does not shows the sensitivity of the controller with the gain parameters.

2.6 Rendezvous problem

Rendezvous refers to the convergence of multiple agents to an agreed state or location through team negotiation to achieve a consensus [29]. The agents may start off at different initial position and navigate using the information available from neighbouring agents. Rendezvous is achieved when all the agents meet at the specified set point and ensure there is a separation to avoid any collision.

Having sensing and connectivity constraints, Kan et. al [30] simulation demonstrates how rendezvous can be achieved for multiple agents with limited information from the environment. The problem assumes that only a few among the multi-agents has the knowledge of destination, which is known as the informed robots while other

agents only uses local state feedback such as position feedback from neighbouring agents or absolute orientation measurement. The control strategy used known as dipolar navigation function is part of artificial potential field based method. Dipolar navigation function is developed for the informed robots to generate nonholonomic trajectory to achieve desired destination with desired orientation.

Masoud in [31], he presented control protocol using nearest neighbour rule to solve rendezvous for multi-agent system. To proof the performance of the consensus protocol, he first implemented the algorithm on single integrator dynamics, followed by simplified model for double integrator dynamics, unmanned ground vehicle (UGV) and unmanned aerial vehicle (UAV). The nearest neighbour rule presented guarantees the convergence of each agents to the consensus point if the agent able to obtain information from at least one neighbour.

In [1, 32], experiment on rendezvous and axial alignment with multiple wheeled mobile robots using the consensus-based design scheme. Consensus protocol is applied onto MASmote which is a differential drive two-wheeled robot to compute based on neighbouring information and actuate on its own. The MASnet platform (Figure 2.2) also has an overhead camera and base station PC that function as a pseudo-GPS to calculate the position and orientation of individual robots. This experiment shows the limitation and challenges to develop a fully distributed system due to the inaccuracy of encoders of robots, that requires a vision-based pose measurement for correction.

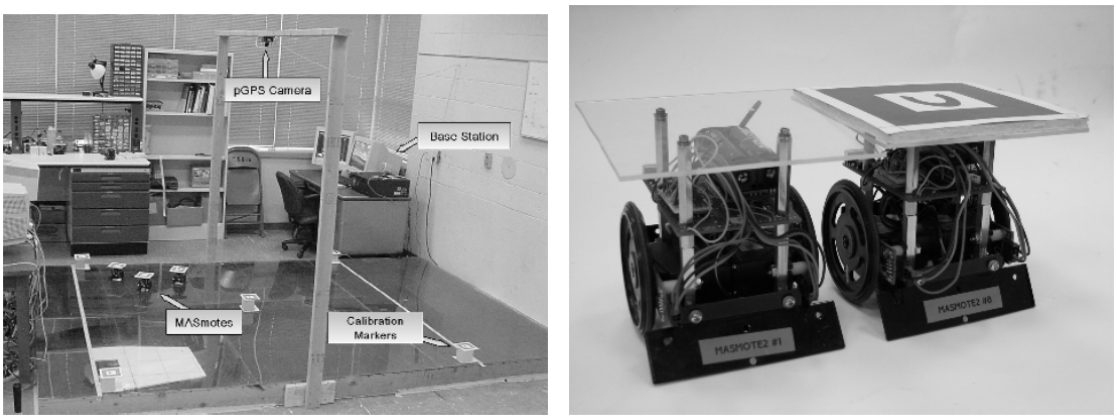


Figure 2.2: (Left) MASnet testbed (Right) MASmote robot hardware [1]

2.7 Graph Theory

An important mathematical tool behind every cooperative control multi-agent system is the graph theory. Suppose that a multi-agent system consists of N individual agents, the network connectivity of agents can be described by graph theory incorporated into the consensus protocol. The system can be described by graph, $G = \{V, E\}$, where $V = \{1, 2, ..., N\}$ represents the node and E is the edge pair between two nodes (i, j). Each node is essentially the individual agents in the system. If there exists an edge between agent i and agent j, this indicates there is a communication between the agents where the state of agent i is available to its neighbouring agent j.

A graph can be further divided into undirected graph and directed graph. For an undirected graph, the communication link between agent i and agent j is bidirectional, where the state of neighbouring agent is available for both. On the other hand, graph that is unidirectional is a directed graph where the state is available for one agent but not the other. Figure 2.3 shows an example of directed graph between two nodes. Note that in most cases, self loop where (i, i) is assumed not allowed.

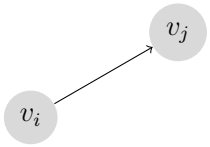


Figure 2.3: Directed graph

Taking advantage of the graph theory mathematical tool, there are several matrices that can be used to describe the network connectivity, which are the adjacency matrix, $A \in \mathbb{R}^{N\times N}$, in-degree matrix, $D \in \mathbb{R}^{N\times N}$ Laplacian matrix, $L \in \mathbb{R}^{N\times N}$. The adjacency matrix $A = [a_{ij}]_{N\times N}$ is defined so that $a_{ij} = 0$ if $(i, j) \notin E$ and $a_{ij} = 1$ if $(i, j) \in E$. The in-degree matrix, D with the node degrees d_i as matrix diagonal elements describe the number of edges going into a node as depicted in Eq. (2.1). Then the Laplacian matrix where $L = D - A$ is given by the Eq. (2.2).

$$D = diag(d_i) = diag\left(\sum_{i=1}^{N} a_{ij}\right) \tag{2.1}$$

$$L = \begin{cases} -a_{ij} & \text{if } i \neq j \text{ and } i \in N \\ \sum_{l \in N} a_{jl} & \text{if } i = j \\ 0 & \text{otherwise} \end{cases} \tag{2.2}$$

2.8 Summary

To conclude for this chapter, some introduction is provided to understand the background of distributed cooperative control with the research progress in mentioned field. Some of the cooperative control problem related to this research, such as formation control, leader following etc. were presented. Most of the research work designed the controller that are guaranteed through Lyapunov function and analysis, but non-that shows how the parameters in the controller would affect the control strategy. Besides, most of the research work only investigated using fixed communication topology between agents which is hard to tell how robust the controller with varying communication topology. Therefore, this research will try to fill in the gap by performing characterization of gain parameters in the chosen controller and evaluate the performance through various communication topology to understand how it would impact on the overall control strategy. The preliminary knowledge on graph theory is provided to provide further understanding in subsequent chapters. The following chapter will discuss the methodology used for this research.

Chapter 3

Methodology

3.1 Introduction

This chapter presents the methodology required to carry out the research. In Section 3.2, the project implementation is described to give a high level overview on this how it is carried out. Section 3.3 provides an introduction to the simulation tool, LabVIEW that is used for this research. In Section 3.3, the kinematic model for wheeled mobile robot required for simulation model is discussed. In Section 3.5, the general coordinate notation for wheeled mobile robot is presented. In Section 3.6, the control objective related to formation control and virtual leader following are discussed here. In Section 3.7, a change of variable required for formation controller is explained. In Section 3.8, the knowledge of graph theory is incorporated to describe the inter-agent connectivity. In Section 3.9, the consensus algorithm required to achieve multi-agent distributed control is presented. Lastly, Section 3.10 described the simulation setup in LabVIEW to carry out the research.

3.2 Project implementation

Figure 3.1 shows the flowchart of the overall project implementation. To begin with, single agent level model is created using the kinematics equation for a non-holonomic wheeled mobile robot. Next, state transformation is used to transform the state in kinematics equation that will be applied in distributed consensus algorithm. Then, the same setup with kinematics model and state transformation is duplicated for four agents. To describe the inter-agent connectivity, a graph matrix generation block generates the respective matrix based on the adjacency matrix information. Control input computation block is constructed using the distributed consensus algorithm

for each agent. The details and theories behind each step will be explained in subsequent subsections.

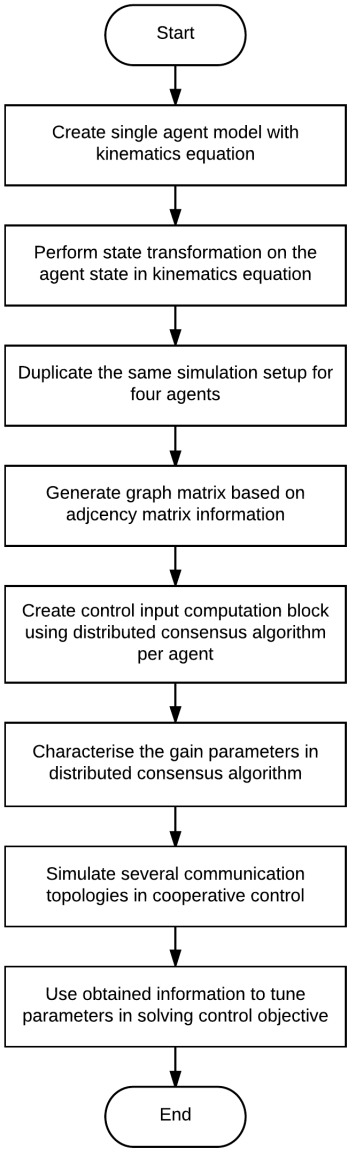


Figure 3.1: Project Flowchart

Once the basic blocks to build the multi-agent system is completed, the first test is to perform characterization on the individual gain parameters in distributed consensus algorithm. Besides, several communication topologies will be simulated to understand the effect of information flow to the cooperative control. Finally, using the characterization results and information from communication topologies, the

parameters in the distributed consensus algorithm is tuned to solve the formation tracking of multi-agent mobile robots.

3.3 Simulation platform

As mentioned in the earlier chapter, this research involves numerical simulation to study distributed control protocol in solving formation control and virtual leader tracking. There are several simulation platforms available for simulation such as Matlab™, Simulink™ and etc. This project attempts to take this opportunity and explore LabVIEW™ in solving academia research problem. Laboratory Virtual Instruments Engineering Workbench (LabVIEW) uses a graphical programming environment that can create user interface, software programming and also various industrial hardware control. The LabVIEW version used in this research is LabVIEW 2015. Besides that, LabVIEW Control Design and Simulation Module is used for control system related simulation. This research utilizes the simulation feature such as memory, integrator, and etc.

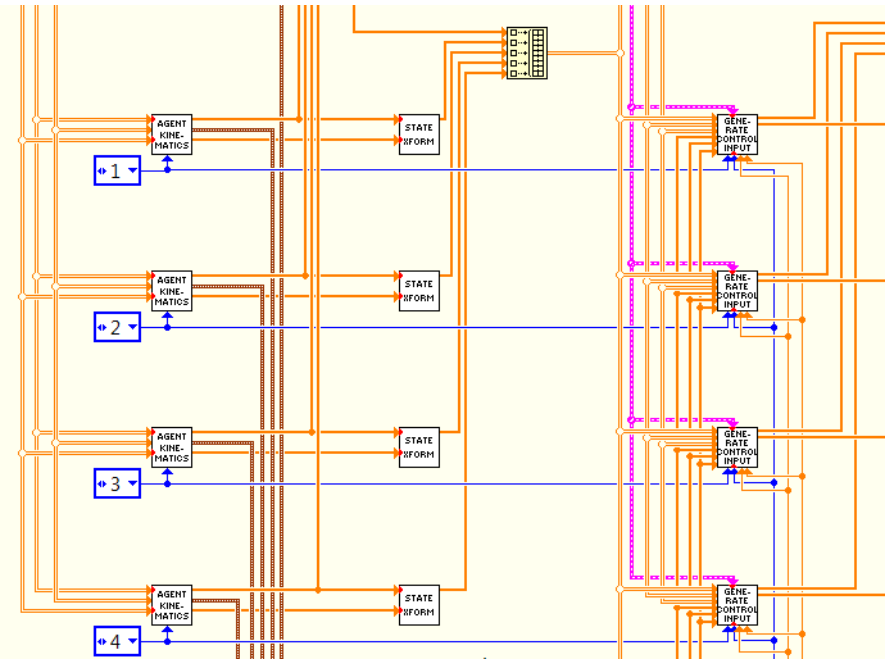


Figure 3.2: Example of subVI in the main VI

There are several benefits in LabVIEW that makes it worth to explore in solving

research problems. Firstly, the graphical programming in LabVIEW can be easily understood rather than using text based programming. LabVIEW also allows user to build smaller programs, also known as subVI which is helpful to simplify a complex VI. Using this allows the simulation files to be more generic, flexible and can be easily modified. Figure 3.2 shows an example of subVI usage in the main simulation VI. The agent kinematics block is created to be generic and simplified so that it can be easily duplicated to simulate more number of agents.

The execution highlighting feature ease the debugging process especially in building complex program. This is very advantageous since there will be multiple agents model to be simulated and things can go complex easily. Even though it is graphical programming environment, formula node feature in LabVIEW also allows mathematical expression to be expressed in terms of text code. An example of simulation loop in LabVIEW that has the mixture between graphical programming and formula node as shown in Figure 3.3.

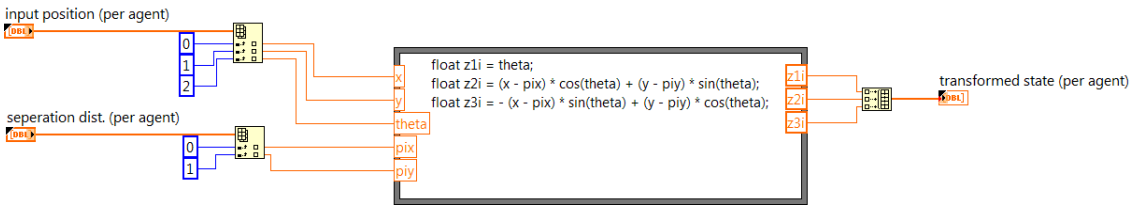
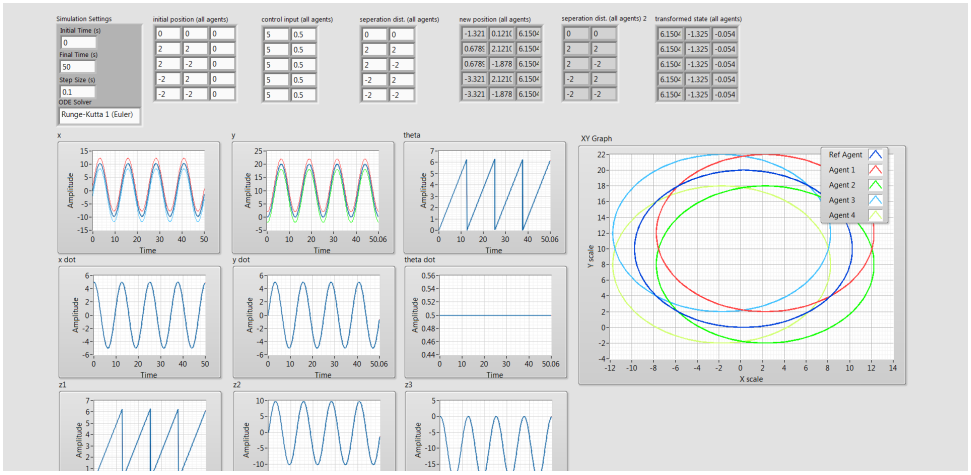


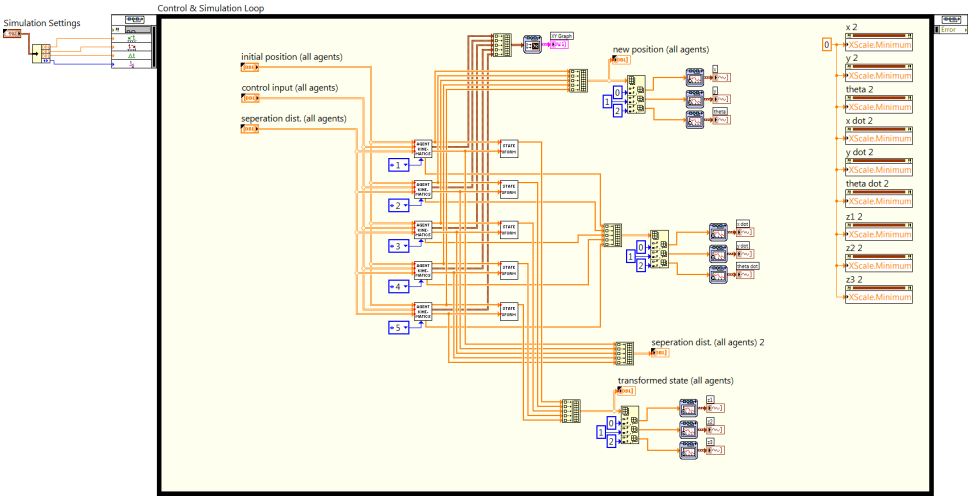
Figure 3.3: Example of formula node in LabVIEW

Lastly, while using LabVIEW, the programming environment allows the user to program both the front panel and block diagram. The front panel is used to develop the user interface of the simulation program. In the front panel, controls are used to obtain user input or simulation settings while indicators are used to display the results. The results from simulation can also be plotted using graphs on front panel. Figure 3.4a shows an example of front panel that contains control and indicators as well as graphs. On the other hand, block diagram is where the actual processing and code is developed. Various LabVIEW built in functions are available for used and subVI can be developed to simplify the overall block diagram. In this research, a main simulation VI is used to establish the framework of simulation. The

cooperative control algorithm is coded into subVI to make it modular and applied to individual agents. Figure 3.4b shows an example of block diagram to simulate agent kinematics.



(a) Front panel

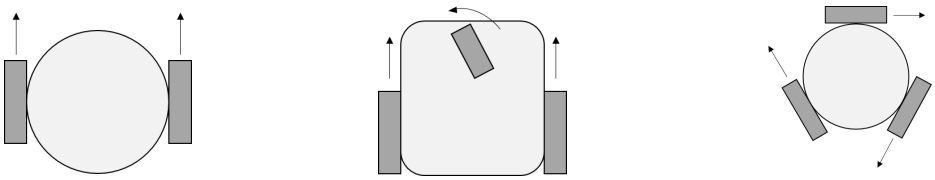


(b) Block diagram

Figure 3.4: Example of front panel and block diagram

3.4 Mobile robot kinematic model

To simulate the cooperative control for a group of nonholonomic mobile robot, it is required to define the model used for single agent level. There are various mobile robot configurations available depending on the degree of manoeuvrability [33, 34]. Some of the common mobile robot configurations are differential drive (Figure 3.5a), tricycle drive (Figure 3.5b), omnidirectional drive (Figure 3.5c) and etc.



(a) Differential Drive (b) Tricycle Drive (c) Omnidirectional Drive

Figure 3.5: Types of mobile robot configurations

Differential drive configuration is considered the simplest driving mechanism for a wheeled mobile robot and will be used as the model for agent robot. For simplicity, the rich nonlinear dynamics of differential drive mobile robot will be excluded to give focus on consensus control. A differential drive robot has two controllable wheels attached to motors and a third free wheel that is used to support the robot platform. The movement of differential drive robot is determined by the relative velocity between the two driving wheels.

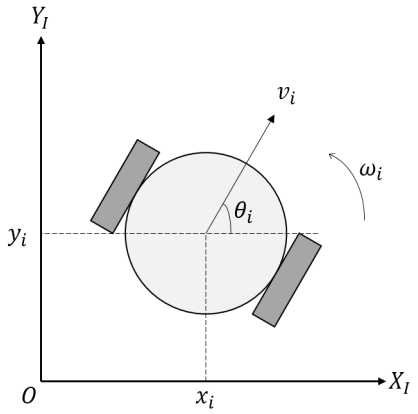


Figure 3.6: Position of robot in Cartesian plane

Figure 3.6 shows a differential drive mobile robot located in a global reference frame, $\{X_I, Y_I\}$. The centre of mass of mobile robot is used to determine the position in the global reference frame. This position can be described by the three variables, x_i, y_i and θ_i. The Cartesian coordinate (x_i, y_i) defines the coordinate of mobile robot in $\{X_I, Y_I\}$ while θ_i defines the heading of mobile robot.

$$\begin{bmatrix} \dot{x_i} \\ \dot{y_i} \\ \dot{\theta_i} \end{bmatrix} = \begin{bmatrix} \cos\theta_i & 0 \\ \sin\theta_i & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} v_i \\ \omega_i \end{bmatrix} \tag{3.1}$$

The corresponding generalised state vector to describe the agent mobile robot is $q_i = [x_i\, y_i\, \theta_i]^T$. Next, the control input of the robot agent is $u_i = [v_i\, \omega_i]^T$ where v_i is the linear velocity and ω_i is the angular velocity of mobile robot. The relationship between the state vector, q_i and control input vector, u_i can be expressed by Eq. (3.1).

Unlike omnidirectional drive, differential drive configuration robot poses non-holonomic constraint. The non-holonomic constraint states that the differential drive robot must satisfy the conditions of pure rolling and non-slipping. In pure rolling condition, the motion of the wheel is in the same direction with motion of robot whereas for non-slipping condition, the wheel motion orthogonal to the wheel plane must be zero. Therefore, these non-holonomic constraints can be expressed by Eq. (3.2).

$$\dot{x_i} \sin\theta i - \dot{y_i} \cos\theta i = 0 \tag{3.2}$$

Using the kinematics equation of mobile robot, the instantaneous position of each mobile robot at time t can be determined using Eq. (3.3) where $[x_i(t)\, y_i(t)\, \theta_i(t)]^T$ is the new state, $[x_i(0)\, y_i(0)\, \theta_i(0)]^T$ is the initial state at $t = 0$. The states of each agents across time and the trajectory on Cartesian plane can be obtained using Eq. (3.3).

$$\begin{bmatrix} x_i(t) \\ y_i(t) \\ \theta_i(t) \end{bmatrix} = \begin{bmatrix} x_i(0) \\ y_i(0) \\ \theta_i(0) \end{bmatrix} + \begin{bmatrix} \int_0^t v_i(t) \cos[\theta_i(t)] dt \\ \int_0^t v_i(t) \sin[\theta_i(t)] dt \\ \int_0^t \omega_i(t) dt \end{bmatrix} \tag{3.3}$$

3.5 Coordinate system

Before going into details of the formation control, it is also worth to discuss on the coordinate system convention used to establish the mobile robot model as it will be implemented in state transformation. Figure 3.7 shows a mobile robot on the global reference frame, $\{X_I, Y_I\}$, having another inertial coordinate frame at the centre

of mass given by $\{X_R, Y_R\}$. The position of this inertial coordinate frame on the global reference frame is given by any arbitrary position, (x_i, y_i) with a rotation of θ_i between the frames.

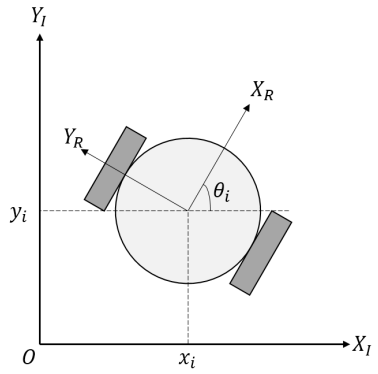


Figure 3.7: Inertial frame $\{X_R, Y_R\}$ on mobile robot

Unlike robot manipulators, the motion of wheeled mobile robot only occurs along the $X - Y$ plane, it is possible to superimpose the inertial coordinate frame and global reference frame with a rotation about the z-axis by an angle, θ_i which is shown in Figure 3.8. Note that the translation of reference frame is excluded, only the rotation of frame is considered in the frame transformation.

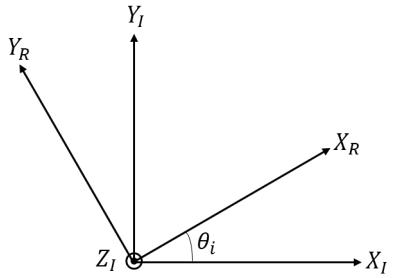


Figure 3.8: Frame rotation between $\{X_R, Y_R\}$ and $\{X_I, Y_I\}$

A rotation matrix, $R(\theta_i)$ can be used to describe the motion of robot on its inertial coordinate frame relative to global reference frame. Eq. (3.4) describes the rotation transformation from $\{X_R, Y_R\}$ to $\{X_I, Y_I\}$.

$$\begin{bmatrix} X_I \\ Y_I \\ \theta_I \end{bmatrix} = \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 \\ \sin\theta_i & \cos\theta_i & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} X_R \\ Y_R \\ \theta_R \end{bmatrix} = R(\theta_i) \cdot \begin{bmatrix} X_R \\ Y_R \\ \theta_R \end{bmatrix} \tag{3.4}$$

Similarly, the motion of mobile robot on the global reference frame can also be transformed back into the inertial reference frame by applying the transposed rotation matrix on the state which is given by Eq. (3.5).

$$\begin{bmatrix} X_R \\ Y_R \\ \theta_R \end{bmatrix} = \begin{bmatrix} \cos\theta_i & \sin\theta_i & 0 \\ -\sin\theta_i & \cos\theta_i & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} X_I \\ Y_I \\ \theta_I \end{bmatrix} = R^T(\theta_i) \cdot \begin{bmatrix} X_I \\ Y_I \\ \theta_I \end{bmatrix} \tag{3.5}$$

3.6 Distributed control objective

In this research, the control objective for distributed control of multi-agent mobile robot is composed of a formation control and virtual leader following. In the formation control, each agents are required to maintain a pre-defined formation among a group of N agents of mobile robot described by a geometric pattern $\mathcal{P}$. Figure 3.9 shows an example of square formation with containment control of the centre agent.

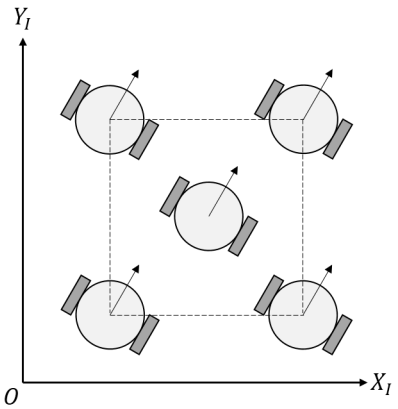


Figure 3.9: Desired formation of mobile robots

The pattern $\mathcal{P}$ is described using an orthogonal coordinate system, $\{P_x, P_y\}$ which has to satisfy the condition given by the Eq.(3.6) where (p_{xi}, p_{yi}) is the position of ith agent and (p_{x0}, p_{y0}) is the centroid of formation. Without the loss of generality, the centroid of formation is defined to be $p_{x0} = p_{y0} = 0$ which will be the origin of the orthogonal coordinate system. Figure 3.10 shows an illustration of orthogonal

coordinate for four agents formation.

$$\sum_{i=1}^{N} p_{xi} = p_{x0} \tag{3.6a}$$

$$\sum_{i=1}^{N} p_{yi} = p_{y0} \tag{3.6b}$$

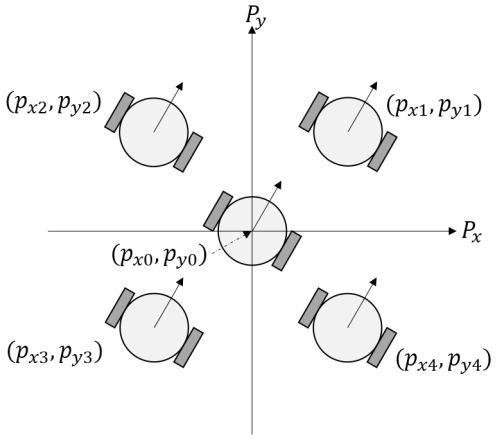


Figure 3.10: Example of orthogonal coordinate for square formation

In order to determine the performance of distributed control law in fulfilment of formation control, the multi-agent mobile robot has to fulfil the condition (3.7), (3.8) and (3.9). Firstly, condition (3.7) states that each agent in the system has to maintain a safe separation among agents defined by the orthogonal coordinates. Secondly, condition (3.8) states that each agent has to converge to the same orientation in the formation. Lastly, condition (3.9) states that the geometric centroid of multi-agent must converge to (p_{x0}, p_{y0}) to achieve the desired formation $\mathcal{P}$.

$$\lim_{t\to\infty} \begin{bmatrix} x_j - x_i \\ y_j - y_i \end{bmatrix} = \begin{bmatrix} p_{xj} - p_{xi} \\ p_{jy} - p_{ji} \end{bmatrix} \tag{3.7}$$

$$\lim_{t\to\infty} (\theta_j - \theta_i) = 0 \tag{3.8}$$

$$\lim_{t\to\infty} \left(\sum_{i=1}^{N} \frac{x_i}{m} - p_{x0} \right) = 0, \lim_{t\to\infty} \left(\sum_{i=1}^{N} \frac{y_i}{m} - p_{y0} \right) = 0 \tag{3.9}$$

Besides formation control, the multi-agent mobile robot also required to perform a virtual leader following at the same time. A time varying trajectory will be

provided to the virtual leader and there must be at least one agent in the system that has this piece of information. The reference trajectory to the virtual leader is given by $q_0 = (x_0(t), y_0(t), \theta_0(t))$ which also satisfy the kinematics given by Eq. (3.10).

$$\begin{bmatrix} \dot{x_0} \\ \dot{y_0} \\ \dot{\theta_0} \end{bmatrix} = \begin{bmatrix} \cos\theta_0 & 0 \\ \sin\theta_0 & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} v_0 \\ \omega_0 \end{bmatrix} \tag{3.10}$$

By introducing the requirement of virtual leader following in addition to formation control, the control objective stated in (3.8) and (3.9) is slightly modified into (3.11) and (3.12). The condition (3.11) states that each of the agent has to orient itself aligned with the virtual leader while condition (3.12) states that the geometric centroid of multi-agent must now converge to the desired trajectory given by (x_0, y_0). Condition (3.7) is still required to ensure the safe separation and to achieve desired formation.

$$\lim_{t\to\infty} (\theta_i - \theta_0) = 0 \tag{3.11}$$

$$\lim_{t\to\infty} \left(\sum_{i=1}^{N} \frac{x_i}{m} - x_0 \right) = 0, \lim_{t\to\infty} \left(\sum_{i=1}^{N} \frac{y_i}{m} - y_0 \right) = 0 \tag{3.12}$$

The control objectives have been described mathematically given by Eq. (3.7), (3.11) and (3.12). Figure 3.11 is used to translate these mathematical conditions into the expected formation given by a four agents system graphically. As it can be seen from the figure, the dark blue line indicates the desired trajectory given by the virtual leader. When the control objective is achieved in steady state, all the four agents must form the square formation as shown by the dotted square, having the virtual leader at the centre of formation. Thus, at any time of the trajectory, this formation must be maintained as well as tracking the trajectory of virtual leader. With clearly statement of the distributed control objective, a formation controller